120

AF/2612

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of	Group Art Unit: 2612
Quan Vu *et al.*	Examiner: Wilson, J.
Serial No. 09/249,642	**TRANSMITTAL LETTER**
Filed: February 12, 1999	162 North Wolfe Road Sunnyvale, California 94086 (408) 530-9700
For: **METHOD OF AND APPARATUS FOR GENERATING A PRECISE FRAME RATE IN DIGITAL VIDEO TRANSMISSION FROM A COMPUTER SYSTEM TO A DIGITAL VIDEO DEVICE**	Customer No. 28960

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Enclosed please find an appeal brief in triplicate including 3 exhibits for filing with the U.S. Patent and Trademark Office. Also attached is a check in the amount of $340.00 for filing the appeal brief.

The Commissioner is authorized to charge any additional fee or credit any overpayment to our Deposit Account No. 08-1275. **An originally executed duplicate of this transmittal is enclosed for this purpose.**

Respectfully submitted,
HAVERSTOCK & OWENS LLP

Dated: October 25, 2004

By: Jonathan O. Owens
Jonathan O. Owens
Reg. No. 37,902

Attorneys for Applicants

CERTIFICATE OF MAILING (37 CFR§ 1.8(a))

I hereby certify that this paper (along with any referred to as being attached or enclosed) is being deposited with the U.S. Postal Service on the date shown below with sufficient postage as first class mail in an envelope addressed to the: Commissioner for Patents, P.O. Box 1450 Alexandria, VA 22313-1450

HAVERSTOCK & OWENS LLP

Date: 10/25/04 By:

PATENT
Atty. Docket No.: SONY-11300

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of	Group Art Unit: 2612
Quan Vu *et al.*	Examiner: Wilson, J.
Serial No. 09/249,642	
Filed: February 12, 1999	**APPEAL BRIEF**
For: **METHOD OF AND APPARATUS FOR GENERATING A PRECISE FRAME RATE IN DIGITAL VIDEO TRANSMISSION FROM A COMPUTER SYSTEM TO A DIGITAL VIDEO DEVICE**	162 N. Wolfe Rd. Sunnyvale, CA 94086 (408) 530-9700
	Customer No. 28960

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In furtherance of patent owner's Notice of Appeal filed on August 27, 2004, an Appeal Brief is submitted herewith in triplicate. This Appeal Brief is written in support of the patent owner's Notice of Appeal filed on August 27, 2004, and further pursuant to the rejection mailed on June 4, 2004.

Claims 1, 2, 4-8 and 10-32 have been rejected. The appellant submits this brief to the Board of Patent Appeals and Interferences in compliance with the requirements of 37 C.F.R. § 1.192. The appellant contends that the rejection of Claims 1, 2, 4-8 and 10-32 in this pending application is in error and should be overcome by this appeal.

I. REAL PARTY IN INTEREST

As the assignees of the entire right, title and interest in the above-captioned patent application, the real parties in interest in this appeal are the following parties:

Sony Corporation, a Japanese Corporation
6-7-35 Kitashinagawa, Shinagawa
Tokyo, 141
Japan

Sony Electronics Inc., a corporation of the State of Delaware
1 Sony Drive
Park Ridge, NJ 07656-8003

per the assignment document recorded on February 12, 1999 at reel number 9780 and frame number 0990.

II. RELATED APPEALS AND INTERFERENCES

There are no other appeals or interferences related to the present patent application of which appellant is aware.

III. STATUS OF CLAIMS

Claims 1, 2, 4-8 and 10-32 are pending within this application. Claims 1, 2, 4-8, 10-20, 23-25 and 28-31 stand rejected under 35 U.S.C. § 102 (e). Claims 21, 22, 26, 27 and 32 stand rejected under 35 U.S.C. § 103 (a).

The rejections of Claims 1, 2, 4-8 and 10-32 are being appealed.

IV. STATUS OF AMENDMENTS

No amendments have been filed subsequent to the Office Action of June 4, 2004. The present condition of the claims is as listed in the Preliminary Amendment filed on April 30, 2004.

V. SUMMARY OF THE INVENTION

The claimed invention is directed to a method of and apparatus for transmitting an isochronous video stream of data at a particular frame rate from a source device to a receiving

device. The receiving device determines the particular, desired frame rate. [Present Specification, page 4, lines 1-2] The source device preferably determines a proper ratio of data packets versus video frames in response to the particular frame rate required and a cycle time for isochronous data. [Present Specification, page 4, lines 2-4] This proper ratio of data packets versus video frames rarely computes to an integer result. [Present Specification, page 4, lines 4-5] Accordingly, once the proper ratio of data packets versus video frames is determined, the source device preferably generates two groups of frames. [Present Specification, page 4, lines 5-6] A first group contains an integer value of packets nearest to and above the desired overall average ratio of data packets versus video frames. [Present Specification, page 4, lines 6-8] The source device also generates a second group of frames where each frame from this second group contains an integer value of packets nearest to and below the ratio of packets versus video frames. [Present Specification, page 4, lines 8-10] In order to achieve the desired frame rate, the source device generates a frame ratio containing a specific number of frames from the first group and the second group and forms the isochronous stream of video data. [Present Specification, page 4, lines 10-12] Accordingly, the frames from the first group and the frames from the second group are of a same type and have the same characteristics.

The source device serially generates each of the frames in an order including a combination of the first group of frames and the second group of frames to achieve the overall desired average frame ratio. [Present Specification, page 4, lines 12-15] The source device then transmits the resulting isochronous video stream of data to the receiving device at the desired frame rate. [Present Specification, pager 4, lines 15-16]

Within the present application, an equation (1) is taught to calculate the number of packets necessary per frame to achieve the required frame rate:

$$\frac{1}{\textit{frame rate}} * \frac{1}{\textit{cycle time}} = \frac{\textit{No. of packets}}{\textit{frame}} \qquad \text{Equation } \mathbf{(1)}$$

[Present Specification, page 9, lines 5-9] For a required frame rate of 29.9700 frames per second and a cycle time of 125 microseconds per cycle, the resulting number of packets per frame is equal to 266.9336 packets per frame. [Present Specification, page 9, lines 11-13] A data stream is then formed from a ratio of frames containing different whole numbers of packets. [Present Specification, page 9, lines 14-18] In order to achieve the overall average of 266.9336 packets per frame over the course of 10,000 frames, 9336 frames are generated containing 267 packets and 664 frames are generated containing 266 packets, yielding a ratio of fourteen frames

containing 267 packets to every one frame containing 266 packets. [Present Specification, page 9, lines 18-22] As taught within the present specification, the originating device generates fourteen frames containing 267 packets followed by one frame containing 266 packets. [Present Specification, page 9, line 24 - page 10, line 2] This pattern is then repeated in order to generate and transmit the data stream. [Present Specification, page 10, lines 2-11] The pattern of fourteen frames from the first group of frames (267 packets) interrupted by one frame from the second group of frames (266 packets) continues as long as the data stream is being transmitted. [Present Specification, page 10, lines 12-22, Figure 6] The calculation of the ratio of frames with 267 packets and frames with 266 packets is only performed once and then the repeating pattern is continued as long as the data stream is being transmitted. This repeating pattern evenly distributes the frames from the second group of frames among the frames from the first group of frames.

VI. ISSUES

The issues presented by the appellant for review by the Board of Patent Appeals and Interferences are as follows:

1. Whether the Claims 1, 2, 4-8, 10-20, 23-25 and 28-31 are properly rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,373,821 to Staats (hereinafter "Staats"); and
2. Whether the Claims 21, 22, 26, 27 and 32 are properly rejected under 35 U.S.C. § 103(a) as being unpatentable over Staats.

VII. GROUPING OF CLAIMS

The claims pending for appeal in the present application do not stand or fall together.

Regarding the rejection of Claims 1, 2, 4-8, 10-20, 23-25 and 28-31 under 35 U.S.C. § 102(e) as being anticipated by Staats, Claims 1, 2, 4-8, 10-20, 23-25 and 28-31 do not stand or fall together because they each contain different limitations. Appellant sets forth, in the argument section of the brief, why the claims are believed to be separately patentable, and therefore should not stand or fall according to the grouping of the claims presented in this rejection. Relevant to this rejection under 35 U.S.C. § 102(e): Claims 1, 2, 4 and 5 can be grouped together; Claims 6-8 and 10-12 can be grouped together; Claims 13-16 can be grouped together; and Claims 17-20 and 23-25 can be grouped together. Claims 28-31 should each stand on their own.

Regarding the rejection of Claims 21, 22, 26, 27 and 32 under 35 U.S.C. § 103(a) as being unpatentable over Staats, Claims 21, 22, 26, 27 and 32 do not stand or fall together because they each contain different limitations. Appellant sets forth, in the argument section of the brief, why the claims are believed to be separately patentable, and therefore should not stand or fall according to the grouping of the claims presented in this rejection.

VIII. ARGUMENT

A. Claims 1, 2, 4-8, 10-20, 23-25 and 28-31 Are Patentable Over Staats

Within the Office Action, Claims 1, 2, 4-8, 10-20, 23-25 and 28-31 have been rejected under 35 U.S.C. § 102(e) as being anticipated by Staats. Staats teaches a method for setting a time stamp in the SYT field of packet headers for IEEE-1394 devices. [Staats, Title] Staats teaches stamping isochronous data packets with a presentation time stamp value determined according to a computed packet rate for the data. [Staats, col. 2, lines 45-48] Staats teaches that a computed packet rate for the data can be a non-integer value. [Staats, col. 5, lines 64-65, col. 6, lines 7-8] To achieve this non-integer value, Staats teaches using a data stream command language. [Staats, col. 6, lines 14-16] The data stream command language is a set of commands that control data flow into or out of a data stream. [Staats, col. 6, lines 16-20] Staats teaches that the data stream command language jump commands are used to allow a transmitter to send a frame with a different number of packets. [Staats, col. 6, lines 27-32] Staats does not teach forming x number of first data blocks each containing n units of data, forming y number of second data blocks each containing m units of data and combining x number of first data blocks and y number of second data blocks into a data stream to achieve the predetermined rate.

Staats teaches that sometimes the transmitter will need to send 266 packets/ frame and sometimes 267 packets/frame. [Staats, col. 6, lines 7-16] Staats teaches that the system determines when the driver should be notified to vary the default number of packets per frame, on a frame by frame basis. [Staats, col. 8, lines 21-67] Specifically, Staats teaches calculating a delta value for each frame, such that if the delta value is equal to or greater than one, a frame with 267 packets is transmitted and if the delta value is less than one, a frame with 266 packets is transmitted. [Staats, col. 8, lines 54-61] In the example, illustrated in Table 1 of Staats, three frames with 266 packets are followed by one frame of 267 packets, then one frame of 266 frames and then one frame of 267 packets. Accordingly, as explicitly shown in this example, Staats does not teach that the frames with 267 packets are evenly distributed among the frames with 266 packets within the data stream, as claimed in the present claims.

Staats teaches calculating an SYT value for a current frame and then calculating the delta value for the current frame, on a frame by frame basis. Staats does not teach evenly distributing x number of first data blocks among y number of second data blocks thereby forming a repeating pattern of the first data blocks and the second data blocks within the data stream. In contrast, as described above, in the present application it is taught that after every fourteen frames from a first group of frames (A), one frame from a second group of frames (B) is inserted. [Present Specification, page 10, lines 12-22, Figure 6] It is further taught in the present application that this repeating pattern of fourteen frames from the first group of frames (A) interrupted by one frame from the second group of frames (B), continues as long as the data stream is transmitted. This is an evenly distributed, repeating pattern of frames. Such an evenly distributed, repeating pattern of frames is not taught by Staats.

Within the Office Action, a position has been taken that Staats teaches that over a period of time, the sequence of data blocks will eventually repeat itself. No where is this taught, hinted at or suggested in the actual teachings of Staats. As discussed above, Staats teaches calculating the delta value for each frame and making a determination based on the delta value as to whether a frame with 267 packets or a frame with 266 packets should be transmitted. Based on this scheme taught by Staats, one cannot **assume** that a pattern will eventually repeat over time, no matter how long the time period. In fact, Staats goes further and even describes what happens when a cycle is missed. This repeating sequence argument, made within the Office Action, does not take such events into account and thus fails when the actual teachings of Staats are analyzed.

Staats teaches that sometimes cycles are missed and a packet is not transmitted every cycle. [Staats, col. 9, line 64- col. 10, line 5] Staats teaches that in the event of a missed cyle "M should be held at 266 to accommodate the missed cycle." [Staats, col. 10, lines 22-23] Staats further teaches that "[t]o accommodate the possibility that missed cycles may occur, the actual cycle # that a frame begins transmission on must be determined and accounted for." [Staats, col. 10, lines 42-44] Thus, Staats teaches that missed cycles must be accounted for and may effect the determined number of packets for a frame or frames in order to maintain the appropriate packet rate.

The position which has been taken within the Office Action regarding an assumed repeating pattern does not follow the teachings of Staats. Staats simply teaches calculating a delta value for each frame and determining on a frame-by-frame basis as to whether a frame with 267 packets or a frame with 266 packets should be transmitted. Staats also teaches that missed cycles must be accounted for in this process. A projected calculation as listed within the Office Action, does not take all of these factors into account, but instead manipulates the data to attempt

to provide a basis for the improper rejection of the pending claims based on Staats. It should be further noted that even this improper projected calculation within the Office Action does not show that frames of the second group are evenly distributed, as taught and claimed in the present application.

Further, as shown even in the example taught by Staats, the frames are not evenly distributed. As described above, in the example taught by Staats, there are three frames with 266 packets, followed by one frame of 267 packets, followed by a single frame with 266 packets and a single frame with 267 packets. Thus, the frames of 267 packets are not evenly distributed among the frames of 266 packets.

There is nothing in the teachings of Staats that supports an anticipation rejection under 35 U.S.C. § 102 of claims with such limitations. Staats simply does not teach evenly distributing the x number of first data blocks among the y number of second data blocks. Staats also does not teach that this even distribution forms a repeating pattern of the first data blocks and the second data blocks within the data stream. As described above, Staats teaches determining the number of packets per frame on a frame by frame basis using a calculated delta value. Also, as described above, the example shown in Table 1 of Staats does not show an even distribution or a repeating pattern, even before missed cycles are accounted for.

As described above, Staats teaches determining, on a frame by frame basis, what number of packets will be included within a frame. The teachings of Staats require this determination to be made for every frame. In contrast to the teachings of Staats, the present invention calculates a ratio of first frames and second frames in response to the particular frame rate. It is taught within the present specification that

> [t]he source device preferably determines a proper ratio of data packets versus video frames in response to the particular frame rate required and a cycle time for isochronous data. This proper ratio of data packets versus video frames rarely computes to an integer result. Accordingly, once the proper ratio of data packets versus video frames is determined, the source device preferably generates two groups of frames. [Present Specification, page 4, lines 2-6]

Staats does not teach calculating a ratio of first frames and second frames. As discussed above, Staats teaches determining a number of packets for each frame, on a frame by frame basis. Further, Staats does not teach calculating a ratio before forming the two groups of frames.

Claim 1

The independent Claim 1 is directed to a method of transmitting information from a source device at a predetermined rate. The method of Claim 1 comprises calculating a ratio of first data blocks to second data blocks to achieve the predetermined rate, forming x number of the first data blocks wherein each of the first data blocks contains n units of data, forming y number of the second data blocks wherein each of the second data blocks contains m units of data, and further wherein m is not equal to n and combining x number of first data blocks and y number of second data blocks into a data stream to achieve the predetermined rate. Claim 1 includes the further limitation that the first data blocks and the second data blocks are of a same type and have same characteristics. Claim 1 also includes the limitation that the x number of first data blocks are **evenly distributed** among the y number of second data blocks thereby forming a repeating pattern of the first data blocks and the second data blocks within the data stream. As described above, Staats does not teach forming x number of first data blocks each containing n units of data, forming y number of second data blocks each containing m units of data and combining x number of first data blocks and y number of second data blocks into a data stream to achieve the predetermined rate. As also described above, Staats does not teach that the x number of first data blocks are **evenly distributed** among the y number of second data blocks thereby forming a repeating pattern of the first data blocks and the second data blocks within the data stream. Further, Staats does not teach **calculating a ratio** of first data blocks to second data blocks to achieve the predetermined rate. As discussed above, Staats teaches determining a number of packets for each frame, on a frame by frame basis. For at least these reasons, the independent Claim 1 is allowable over the teachings of Staats.

Claims 2, 4 and 5

Claims 2, 4 and 5 are all dependent upon the independent Claim 1. As discussed above, the independent Claim 1 is allowable over the teachings of Staats. Accordingly, Claims 2, 4 and 5 are all also allowable as being dependent upon an allowable base claim.

Claim 6

The independent Claim 6 is directed to a method of transmitting information from a source device to a receiving device. The method of Claim 6 comprises calculating a ratio of first frames to second frames to achieve a predetermined frame rate, forming x number of the first frames wherein each of the first frames contains n units of data, forming y number of the second frames wherein each of the second frames contains m units of data and further wherein m is not

equal to n, combining x number of the first frames and y number of the second frames into a stream of frames to achieve the predetermined frame rate by **evenly distributing** the x number of the first frames among the y number of the second frames thereby forming a repeating pattern of the first frames and the second frames within the stream of frames and transmitting the stream of frames from the source device to the receiving device. Claim 6 includes the further limitation that the first frames and the second frames are of a same type and have same characteristics. As described above, Staats does not teach forming x number of first frames wherein each of the first frames contains n units of data, forming y number of second frames wherein each of the second frames contains m units of data and combining x number of the first frames and y number of the second frames into a stream of frames to achieve a predetermined rate. As discussed above, Staats also does not teach **evenly distributing** the x number of first frames among the y number of second frames thereby forming a repeating pattern of the first frames and the second frames within the stream of frames. Further, Staats does not teach **calculating a ratio** of first frames to second frames to achieve a predetermined frame rate. As discussed above, Staats teaches determining a number of packets for each frame, on a frame by frame basis. For at least these reasons, the independent Claim 6 is allowable over the teachings of Staats.

<u>Claims 7, 8 and 10-12</u>

Claims 7, 8 and 10-12 are all dependent upon the independent Claim 6. As discussed above, the independent Claim 6 is allowable over the teachings of Staats. Accordingly, Claims 7, 8 and 10-12 are each also allowable as being dependent upon an allowable base claim.

<u>Claim 13</u>

The independent Claim 13 is directed to a source device for transmitting information at a predetermined frame rate. The source device of Claim 13 comprises a controller for calculating a ratio of first frames to second frames to achieve the predetermined frame rate and generating a data stream containing a plurality of the first frames each including x packets of data and a plurality of the second frames each including y packets of data to achieve the predetermined frame rate, wherein the data stream is transmitted at the predetermined frame rate and y is not equal to x. Claim 13 includes the further limitation that the first frames and the second frames are of a same type and have same characteristics. It is also specified in Claim 13 that the x number of first data blocks are **evenly distributed** among the y number of second data blocks thereby forming a repeating pattern of the first frames and the second frames within the data stream. As described above, Staats does not teach generating a data stream including a plurality

of first frames each including x packets of data and a plurality of second frames each including y packets of data to achieve the predetermined frame rate. As also described above, Staats does not teach that the x number of first data blocks are **evenly distributed** among the y number of second data blocks thereby forming a repeating pattern of the first frames and the second frames within the data stream. Further, Staats does not teach **calculating a ratio** of first frames to second frames to achieve a predetermined frame rate. As discussed above, Staats teaches determining a number of packets for each frame, on a frame by frame basis. For at least these reasons, the independent Claim 13 is allowable over the teachings of Staats.

Claims 14-16

Claims 14-16 are all dependent upon the independent Claim 13. As discussed above, the independent Claim 13 is allowable over the teachings of Staats. Accordingly, Claims 14-16 are each also allowable as being dependent upon an allowable base claim.

Claim 17

The independent Claim 17 is directed to a system for transmitting information at a predetermined frame rate. The system of Claim 17 comprises a source device for calculating a ratio of first frames to second frames to achieve the predetermined frame rate and generating a data stream containing a plurality of first frames each including x packets of data and a plurality of second frames each including y packets of data to achieve the predetermined frame rate and y is not equal to x, wherein the first frames and the second frames are of a same type and have same characteristics and the x number of first data blocks are **evenly distributed** among the y number of second data blocks thereby forming a repeating pattern of the first frames and the second frames within the data stream, and a remote receiver coupled to the source device and configured to receive the data stream at the predetermined frame rate. As described above, Staats does not teach generating a data stream containing a plurality of first frames each including x packets of data and a plurality of second frames each including y packets of data to achieve the predetermined frame rate and y is not equal to x. As discussed above, Staats also does not teach that the x number of first data blocks are **evenly distributed** among the y number of second data blocks thereby forming a repeating pattern of the first frames and the second frames within the data stream. Further, Staats does not teach **calculating a ratio** of first frames to second frames to achieve a predetermined frame rate. As discussed above, Staats teaches determining a number of packets for each frame, on a frame by frame basis. For at least these reasons, the independent Claim 17 is allowable over the teachings of Staats.

Claims 18-20 and 23-25

Claims 18-20 and 23-25 are all dependent on the independent Claim 17. As discussed above, the independent Claim 17 is allowable over the teachings of Staats. Accordingly, Claims 18-20 and 23-25 are each also allowable as being dependent upon an allowable base claim.

Claim 28

Claim 28 is dependent on the independent Claim 1 and adds a further limitation specifying that the predetermined rate is determined by a receiving device which receives the data stream. As described above, it is taught within the present application that the receiving device determines the particular, desired frame rate. [Present Specification, page 4, lines 1-2] Staats does not teach that the receiving device determines the predetermined rate. For at least these reasons, the Claim 28 is allowable over the teachings of Staats.

As an additional basis for allowability, Claim 28 is dependent on the independent Claim 1. As discussed above, the independent Claim 1 is allowable over the teachings of Staats. Accordingly, Claim 28 is also allowable as being dependent upon an allowable base claim.

Claim 29

Claim 29 is dependent on the independent Claim 6 and adds a further limitation specifying that the predetermined rate is determined by the receiving device. As described above, it is taught within the present application that the receiving device determines the particular, desired frame rate. [Present Specification, page 4, lines 1-2] Staats does not teach that the receiving device determines the predetermined rate. For at least these reasons, the Claim 29 is allowable over the teachings of Staats.

As an additional basis for allowability, Claim 29 is dependent on the independent Claim 6. As discussed above, the independent Claim 6 is allowable over the teachings of Staats. Accordingly, Claim 29 is also allowable as being dependent upon an allowable base claim.

Claim 30

Claim 30 is dependent on the independent Claim 13 and adds a further limitation specifying that the predetermined rate is determined by a receiving device which receives the data stream. As described above, it is taught within the present application that the receiving device determines the particular, desired frame rate. [Present Specification, page 4, lines 1-2] Staats does not teach that the receiving device determines the predetermined rate. For at least these reasons, the Claim 30 is allowable over the teachings of Staats.

As an additional basis for allowability, Claim 30 is dependent on the independent Claim 13. As discussed above, the independent Claim 13 is allowable over the teachings of Staats. Accordingly, Claim 30 is also allowable as being dependent upon an allowable base claim.

Claim 31

Claim 31 is dependent on the independent Claim 17 and adds a further limitation specifying that the predetermined rate is determined by the remote receiver. As described above, it is taught within the present application that the receiving device determines the particular, desired frame rate. [Present Specification, page 4, lines 1-2] Staats does not teach that the receiving device determines the predetermined rate. For at least these reasons, the Claim 31 is allowable over the teachings of Staats.

As an additional basis for allowability, Claim 31 is dependent on the independent Claim 17. As discussed above, the independent Claim 17 is allowable over the teachings of Staats. Accordingly, Claim 31 is also allowable as being dependent upon an allowable base claim.

B. Claims 21, 22, 26, 27 and 32 Are Patentable Over Staats

Within the Office Action, Claims 21, 22, 26, 27 and 32 have been rejected under 35 U.S.C. §103 (a) as being unpatentable over Staats.

Claims 21 and 22

Claims 21 and 22 are both dependent on the independent Claim 17. As discussed above, the independent Claim 17 is allowable over the teachings of Staats. Accordingly, Claims 21 and 22 are both also allowable as being dependent upon an allowable base claim.

Claim 26

The independent Claim 26 is directed to a system for transmitting information at a predetermined frame rate equal to 29.97 frames per second within an IEEE 1394 network of devices. The system of Claim 26 comprises a source device for calculating a ratio of first frames to second frames to achieve the predetermined frame rate and generating a data stream containing 9336 first frames, each including 267 packets of data, and 664 second frames, each including 266 packets of data, to achieve the predetermined frame rate of 29.97 frames per second, wherein the first frames and the second frames are of a same type and have same characteristics and the x number of first data blocks are **evenly distributed** among the y number of second data blocks

thereby forming a repeating pattern of first frames and second frames within the data stream, and a remote receiver coupled to the source device by the IEEE 1394 network of devices, wherein the remote receiver receives the data stream from the source device at the predetermined frame rate. As recognized with the Office Action, Staats fails to disclose a data stream containing 9336 first frames and 664 second frames. It is stated in the Office Action that this is an obvious matter of design choice. The applicants respectfully disagree.

Staats cites an NTSC compatible device with 266.973 data packets per frame, as an example. [Staats, col. 5, line 64 - col. 6, line 12] However, as discussed above, Staats only teaches that sometimes the transmitter will need to send 266 packets/frame and sometimes 267 packets/frame. As evidence that the limitation of a data stream containing 9336 first frames, each including 267 packets of data, and 664 second frames, each including 266 packets of data, is not an obvious design choice, even though Staats cites as an example an NTSC compatible device with 266.973 data packets per frame, Staats does not describe such a data stream with 9336 first frames and 664 second frames. Even the projected calculation within the Office Action, which uses 266.973 and portends to follow the teachings of Staats, does not arrive at a data stream with 9336 first frames and 664 second frames.

As discussed above, Staats only teaches that sometimes the transmitter will need to send 266 packets/frame and sometimes 267 packets/frame. Accordingly, Staats does not teach or make obvious a source device for generating a data stream containing 9336 first frames, each including 267 packets of data, and 664 second frames, each including 266 packets of data. As discussed above, Staats also does not teach or make obvious **evenly distributing** the x number of first frames among the y number of second frames thereby forming a repeating pattern of first frames and second frames within the data stream. Further, Staats does not teach **calculating a ratio** of first frames to second frames to achieve a predetermined frame rate. As discussed above, Staats teaches determining a number of packets for each frame, on a frame by frame basis. For at least these reasons, the independent Claim 26 is allowable over the teachings of Staats.

Claim 27

The independent Claim 27 is directed to a method of transmitting information from a source device to a receiving device over an IEEE 1394 network of devices. The method of Claim 27 comprises calculating a ratio of first frames to second frames, forming 9336 first frames wherein each of the first frames contains 267 packets of data, forming 664 second frames wherein each of the second frames contains 266 packets of data, combining the 9336 first frames and the 664 second frames into a stream of frames to achieve a predetermined frame rate of 29.97

frames per second by **evenly distributing** the second frames among the first frames thereby forming a repeating pattern of the first frames and the second frames within the stream of frames and transmitting the stream of frames from the source device to the receiving device over the IEEE 1394 network of devices, wherein the first frames and the second frames are of a same type and have same characteristics. As described above, Staats does not teach or make obvious forming 9336 first frames wherein each of the first frames contains 267 packets of data, forming 664 second frames wherein each of the second frames contains 266 packets of data and combining the 9336 first frames and the 664 second frames into a stream of frames to achieve a predetermined frame rate of 29.97 frames per second. As also described above, even the projected calculation within the Office Action, which uses 266.973 and portends to follow the teachings of Staats, does not arrive at a data stream with 9336 first frames and 664 second frames.

As described above, Staats does not teach **evenly distributing** the second frames among the first frames thereby forming a repeating pattern of the first frames and the second frames within the stream of frames. Further, Staats does not teach **calculating a ratio** of first frames to second frames to achieve a predetermined frame rate. As discussed above, Staats teaches determining a number of packets for each frame, on a frame by frame basis. For at least these reasons, the independent Claim 27 is allowable over the teachings of Staats.

Claim 32

Claim 32 is dependent on the independent Claim 26 and adds a further limitation specifying that the predetermined rate is determined by the remote receiver. As described above, it is taught within the present application that the receiving device determines the particular, desired frame rate. [Present Specification, page 4, lines 1-2] Staats does not teach that the receiving device determines the predetermined rate. For at least these reasons, the Claim 32 is allowable over the teachings of Staats.

As an additional basis for allowability, Claim 32 is dependent on the independent Claim 26. As discussed above, the independent Claim 26 is allowable over the teachings of Staats. Accordingly, Claim 32 is also allowable as being dependent upon an allowable base claim.

C. CONCLUSION

It is therefore respectfully submitted that Claims 1, 2, 4-8 and 10-32 are allowable over the teachings of Staats. Therefore, a favorable indication is respectfully requested.

IX. APPENDIX

Claims Under Appeal

1. A method of transmitting information from a source device at a predetermined rate, the method comprising:
   a. calculating a ratio of first data blocks to second data blocks to achieve the predetermined rate;
   b. forming x number of the first data blocks wherein each of the first data blocks contains n units of data;
   c. forming y number of the second data blocks wherein each of the second data blocks contains m units of data, and further wherein m is not equal to n; and
   d. combining x number of first data blocks and y number of second data blocks into a data stream to achieve the predetermined rate, wherein the first data blocks and the second data blocks are of a same type and have same characteristics and further wherein the x number of first data blocks are evenly distributed among the y number of second data blocks thereby forming a repeating pattern of the first data blocks and the second data blocks within the data stream.

2. The method according to claim 1 further comprising transmitting the data stream from the source device at the predetermined rate.

3. (canceled)

4. The method according to claim 1 wherein the data stream is digital video data.

5. The method according to claim 1 wherein n, m, x, and y are integer values.

6. A method of transmitting information from a source device to a receiving device, the method comprising:

   a. calculating a ratio of first frames to second frames to achieve a predetermined frame rate;

   b. forming x number of the first frames wherein each of the first frames contains n units of data;

   c. forming y number of the second frames wherein each of the second frames contains m units of data, and further wherein m is not equal to n;

   d. combining x number of the first frames and y number of the second frames into a stream of frames to achieve the predetermined frame rate by evenly distributing the x number of the first frames among the y number of the second frames thereby forming a repeating pattern of the first frames and the second frames within the stream of frames; and

   e. transmitting the stream of frames from the source device to the receiving device;

wherein the first frames and the second frames are of a same type and have same characteristics.

7. The method according to claim 6 wherein n, m, x, and y are integer values.

8. The method according to claim 6 further comprising receiving the stream of frames from the network by the receiver at a predetermined frame rate and wherein the data stream conforms to standards of an IEEE 1394-1995 network.

9. (canceled)

10. The method according to claim 6 wherein the stream of frames conforms to standards of an IEEE 1394-1995 network.

11. The method according to claim 6 wherein the source device and the receiving device are coupled together within a network.

12. The method according to claim 11 wherein the network is an IEEE 1394-1995 network.

13. A source device for transmitting information at a predetermined frame rate, the source device comprising a controller for calculating a ratio of first frames to second frames to achieve the predetermined frame rate and generating a data stream containing a plurality of the first frames each including x packets of data and a plurality of the second frames each including y packets of data to achieve the predetermined frame rate, wherein the data stream is transmitted at the predetermined frame rate and y is not equal to x and further wherein the first frames and the second frames are of a same type and have same characteristics and the x number of first data blocks are evenly distributed among the y number of second data blocks thereby forming a repeating pattern of the first frames and the second frames within the data stream.

14. The source device according to claim 13 wherein x and y are integer values.

15. The source device according to claim 13 further comprising an interface coupled to the controller and configured for connecting to a network.

16. The source device according to claim 15 wherein the network is a IEEE 1394-1995 network.

17. A system for transmitting information at a predetermined frame rate, the system comprising:

    a. a source device for calculating a ratio of first frames to second frames to achieve the predetermined frame rate and generating a data stream containing a plurality of the first frames each including x packets of data and a plurality of the second frames each including y packets of data to achieve the predetermined frame rate and y is not equal to x, wherein the first frames and the second frames are of a same type and have same characteristics and the x number of first data blocks are evenly distributed among the y number of second data blocks thereby forming a repeating pattern of the first frames and the second frames within the data stream; and

    b. a remote receiver coupled to the source device and configured to receive the data stream at the predetermined frame rate.

18. The system according to claim 17 wherein x and y are integer values.

19. The system according to claim 17 wherein the source device is a computer system.

20. The system according to claim 17 wherein the remote receiver is a digital video camera.

21. The system according to claim 17 wherein the predetermined frame rate is 29.97 frames per second.

22. The system according to claim 17 wherein the plurality of first frames are 9336 frames, x packets represent 267 packets, the plurality of second frames are 664 frames, and y packets represent 266 packets.

23. The system according to claim 17 wherein the data stream conforms to standards of an IEEE 1394-1995 network.

24. The system according to claim 17 further comprising a network coupled between the source device and the remote receiver and configured to transmit the data stream.

25. The system according to claim 24 wherein the network is an IEEE 1394-1995 network.

26. A system for transmitting information at a predetermined frame rate equal to 29.97 frames per second within an IEEE 1394 network of devices, the system comprising:

  a. a source device for calculating a ratio of first frames to second frames to achieve the predetermined frame rate and generating a data stream containing 9336 first frames, each including 267 packets of data, and 664 second frames, each including 266 packets of data, to achieve the predetermined frame rate of 29.97 frames per second, wherein the first frames and the second frames are of a same type and have same characteristics and the x number of first data blocks are evenly distributed among the y number of second data blocks thereby forming a repeating pattern of first frames and second frames within the data stream; and

  b. a remote receiver coupled to the source device by the IEEE 1394 network of devices, wherein the remote receiver receives the data stream from the source device at the predetermined frame rate.

27. A method of transmitting information from a source device to a receiving device over an IEEE 1394 network of devices, the method comprising:

  a. calculating a ratio of first frames to second frames;

b. forming 9336 first frames wherein each of the first frames contains 267 packets of data;

c. forming 664 second frames wherein each of the second frames contains 266 packets of data;

d. combining the 9336 first frames and the 664 second frames into a stream of frames to achieve a predetermined frame rate of 29.97 frames per second by evenly distributing the second frames among the first frames thereby forming a repeating pattern of the first frames and the second frames within the stream of frames; and

e. transmitting the stream of frames from the source device to the receiving device over the IEEE 1394 network of devices;

wherein the first frames and the second frames are of a same type and have same characteristics.

28. The method according to claim 1 wherein the predetermined rate is determined by a receiving device which receives the data stream.

29. The method according to claim 6 wherein the predetermined frame rate is determined by the receiving device.

30. The source device according to claim 13 wherein the predetermined rate is determined by a receiving device which receives the data stream.

31. The system according to claim 17 wherein the predetermined frame rate is determined by the remote receiver.

32. The system according to claim 26 wherein the predetermined frame rate is determined by the remote receiver.

X. ATTACHMENTS

The following documents, which are part of the record, are attached for convenience:

1. U.S. Patent No. 6,373,821 to Staats.
2. The June 4, 2004 Office Action.
3. The April 30, 2004 Preliminary Amendment.

Respectfully submitted,
HAVERSTOCK & OWENS LLP

Dated: October 25, 2004

By: Jonathan O. Owens
Jonathan O. Owens
Reg. No. 37,902
Attorneys for Applicants

CERTIFICATE OF MAILING (37 CFR§ 1.8(a))

I hereby certify that this paper (along with any referred to as being attached or enclosed) is being deposited with the U.S. Postal Service on the date shown below with sufficient postage as first class mail in an envelope addressed to the: Commissioner for Patents, P.O. Box 1450 Alexandria, VA 22313-1450

HAVERSTOCK & OWENS LLP.

Date: 10/25/04 By:

US006373821B2

(12) **United States Patent**
Staats

(10) **Patent No.: US 6,373,821 B2**
(45) **Date of Patent: *Apr. 16, 2002**

(54) **METHOD FOR SETTING TIME STAMP IN SYT FIELD OF PACKET HEADERS FOR IEEE-1394 DEVICES**

(75) Inventor: **Erik P. Staats**, Ben Lomand, CA (US)

(73) Assignee: **Apple Computer, Inc.**, Cupertino, CA (US)

(*) Notice: This patent issued on a continued prosecution application filed under 37 CFR 1.53(d), and is subject to the twenty year patent term provisions of 35 U.S.C. 154(a)(2).

Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/027,499**

(22) Filed: **Feb. 20, 1998**

(51) **Int. Cl.**[7] **H04J 3/06**; H04J 3/14

(52) **U.S. Cl.** **370/252**; 370/503; 370/394; 348/513

(58) **Field of Search** 370/252, 394, 370/465, 503, 517; 714/798, 812, 818; 348/461, 464, 467, 469, 425.2, 425.3, 425.4, 512, 513

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,506,358 A	* 3/1985	Montgomery	370/409
4,748,620 A	* 5/1988	Adelmann et al.	370/394
4,894,823 A	* 1/1990	Adelmann et al.	370/252
5,127,000 A	* 6/1992	Henrion	370/394
5,319,360 A	* 6/1994	Schrodi et al.	370/390
5,469,466 A	* 11/1995	Chu	375/354
5,483,523 A	* 1/1996	Nederlof	370/409
5,565,924 A	* 10/1996	Haskell et al.	370/503
5,590,116 A	* 12/1996	Zhang	370/253
5,661,728 A	* 8/1997	Finotello et al.	370/503
5,668,601 A	* 9/1997	Okada et al.	370/503
5,682,384 A	* 10/1997	Zarros	370/394
5,726,989 A	* 3/1998	Dokic	370/517
5,742,623 A	* 4/1998	Nuber et al.	714/798
5,802,057 A	* 9/1998	Duckwall et al.	370/408
5,838,876 A	* 11/1998	Iwamura	370/503
5,845,152 A	* 12/1998	Anderson et al.	395/872
5,878,032 A	* 3/1999	Mirek et al.	370/252
5,883,891 A	* 3/1999	Williams et al.	370/356
5,912,880 A	* 6/1999	Bernstein	370/252
5,930,480 A	* 7/1999	Staats	709/244
5,973,748 A	* 10/1999	Horiguchi et al.	370/465
6,006,270 A	* 12/1999	Kobunaya	709/233

OTHER PUBLICATIONS

"P1394 Standard For A High Performance Serial Bus", *The Institute of Electrical and Electronic Engineers, Inc., IEEE Standards Department*, P1394 Draft 8.0v3, pp. 1–394 (Oct. 16, 1995).

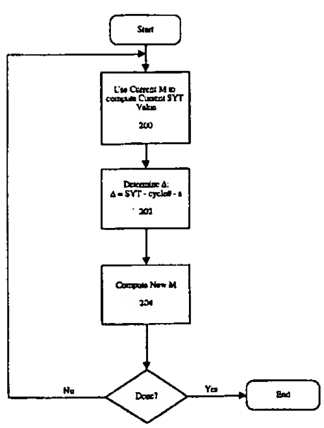
* cited by examiner

Primary Examiner—Chau Nguyen
Assistant Examiner—Phuongchau Ba Nguyen
(74) *Attorney, Agent, or Firm*—Blakely, Sokoloff, Taylor & Zafman LLP

(57) **ABSTRACT**

Isochronous data packets transmitted within a digital network having a bus architecture that complies with the IEEE-1394 Standard for a High Performance Serial Bus are stamped with a presentation time stamp value determined according to a computed packet rate for the data. For the case where the presentation time stamp field of a first packet of a second frame of data for transmission in the digital network is set with the presentation time value, the packet rate may be computed by measuring a difference between a desired presentation time value of a first packet in a first frame of the data and an actual transmission time of the first packet of the first frame. The first frame preceding the second frame in time of transmission within the network.

5 Claims, 6 Drawing Sheets



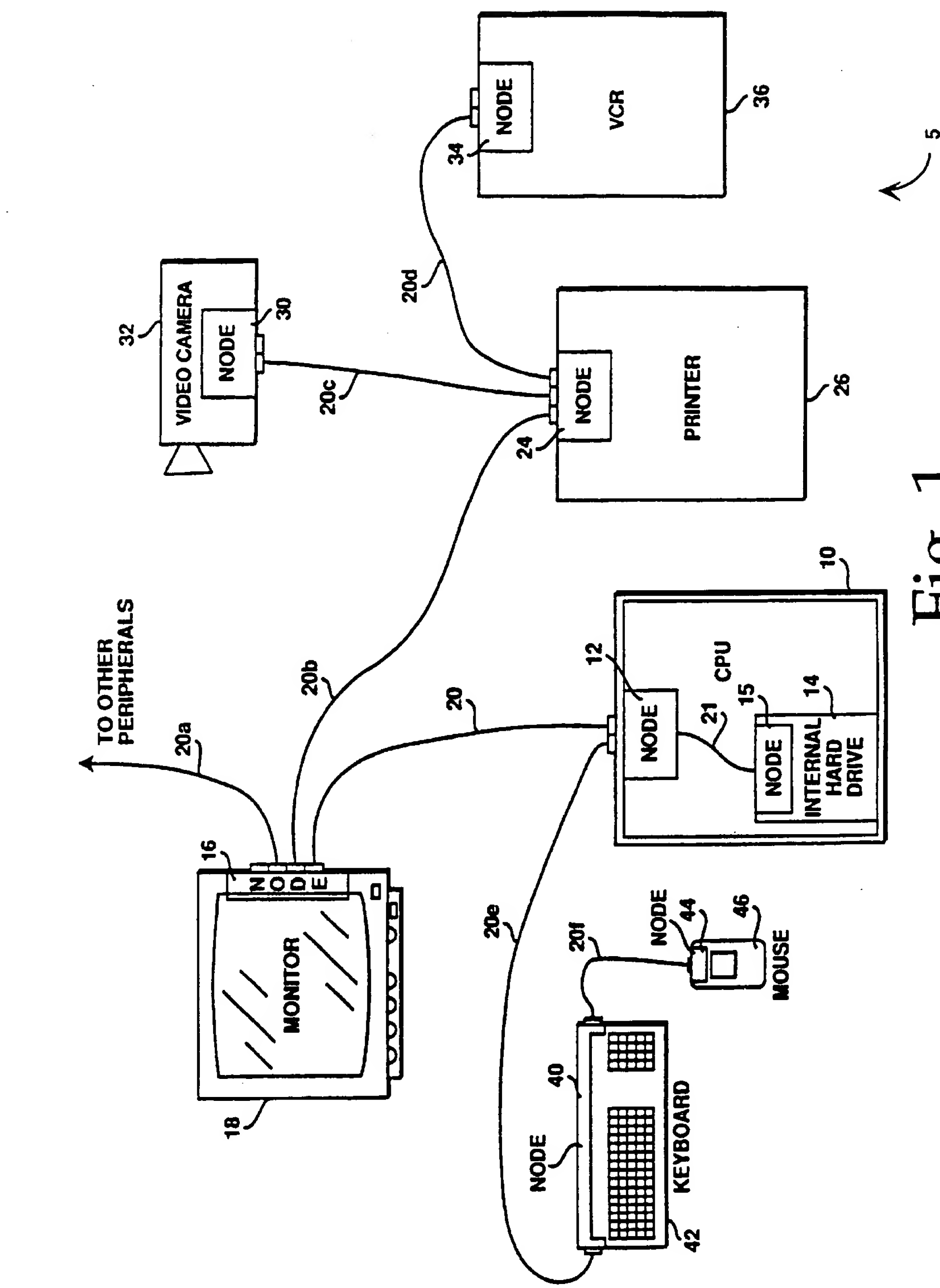

VCR
NODE
36
34
5
20d
VIDEO CAMERA
NODE
32
30
20c
PRINTER
NODE
24
26
TO OTHER PERIPHERALS
20a
20b
20
10
CPU
NODE
12
21
15
14
NODE
INTERNAL HARD DRIVE
16
NODE
MONITOR
18
20e
20f
NODE
44
46
MOUSE
40
NODE
KEYBOARD
42

Fig. 1

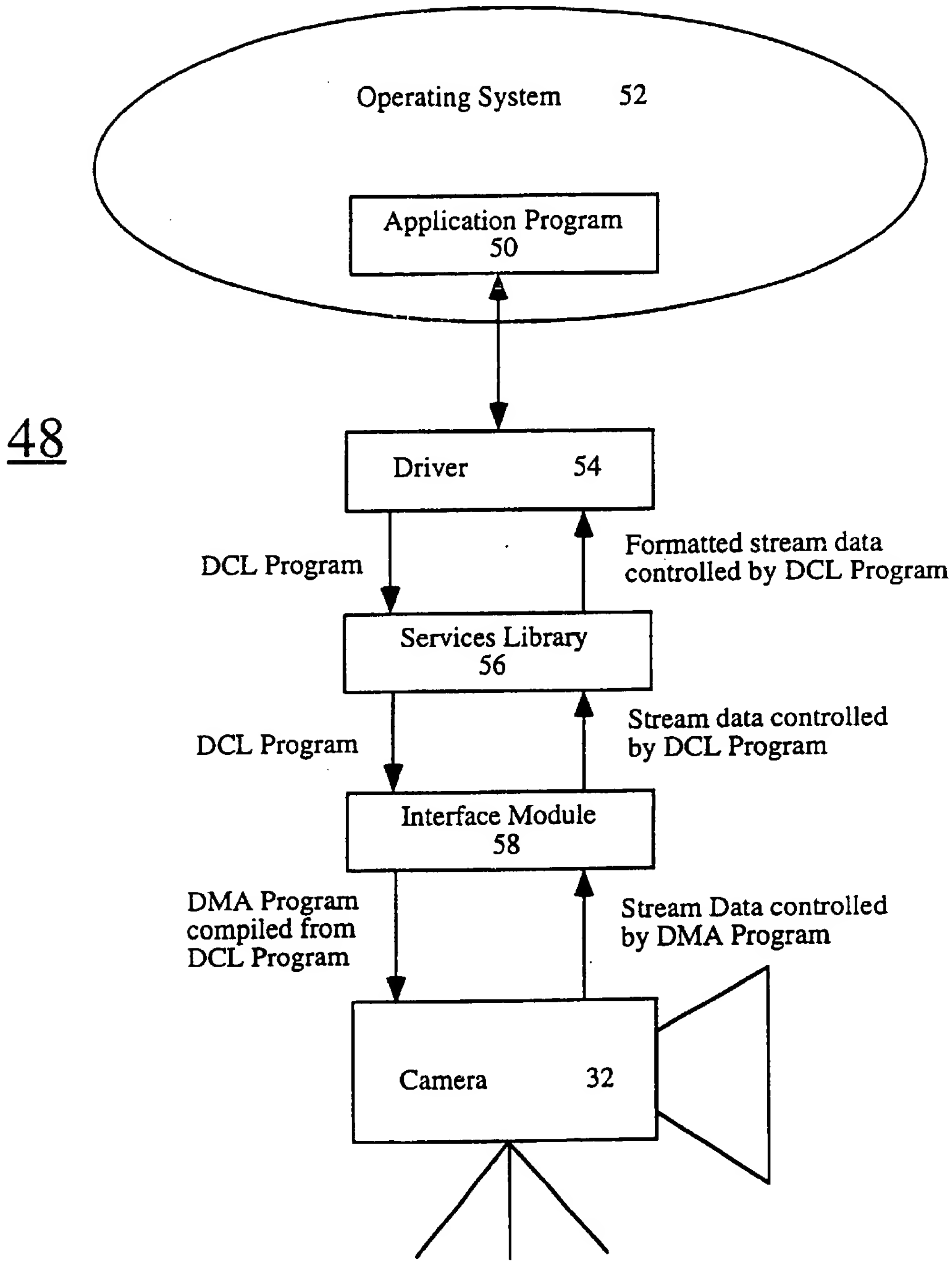

Operating System 52
Application Program
50
48
Driver 54
DCL Program
Formatted stream data
controlled by DCL Program
Services Library
56
DCL Program
Stream data controlled
by DCL Program
Interface Module
58
DMA Program
compiled from
DCL Program
Stream Data controlled
by DMA Program
Camera 32

Fig. 2

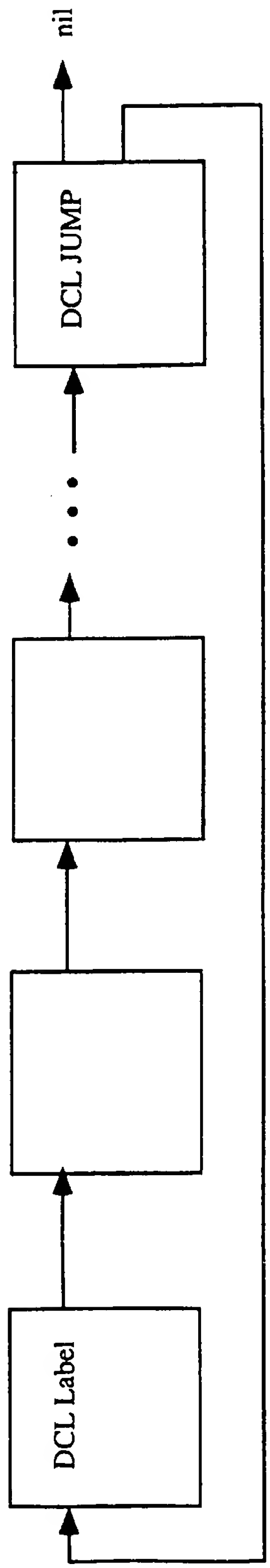


Fig. 3

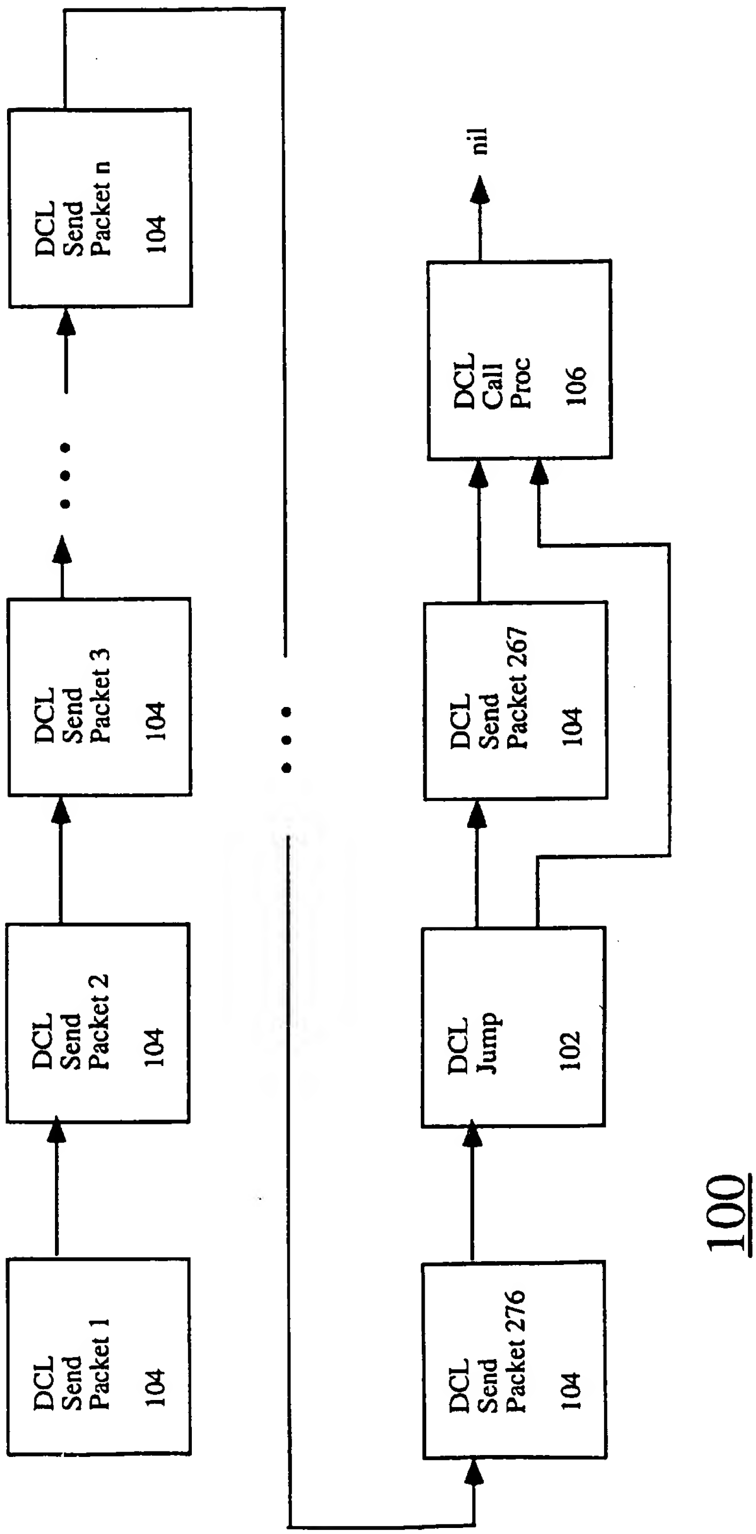

DCL Send Packet 1 104
DCL Send Packet 2 104
DCL Send Packet 3 104
DCL Send Packet n 104
DCL Send Packet 276 104
DCL Jump 102
DCL Send Packet 267 104
DCL Call Proc 106
nil
100

Fig. 4

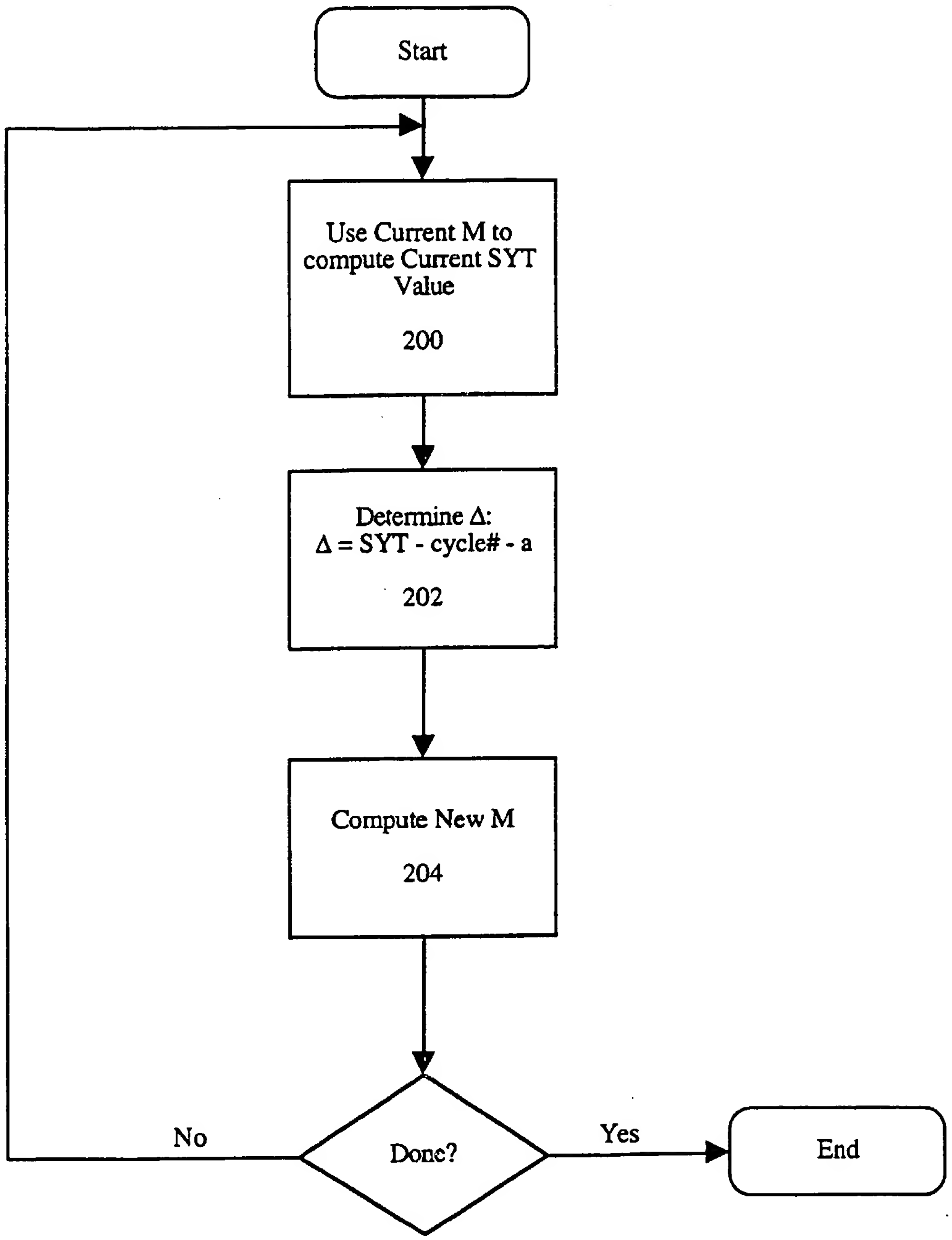

Start
Use Current M to compute Current SYT Value
200
Determine Δ:
Δ = SYT - cycle# - a
202
Compute New M
204
No
Done?
Yes
End

Fig. 5

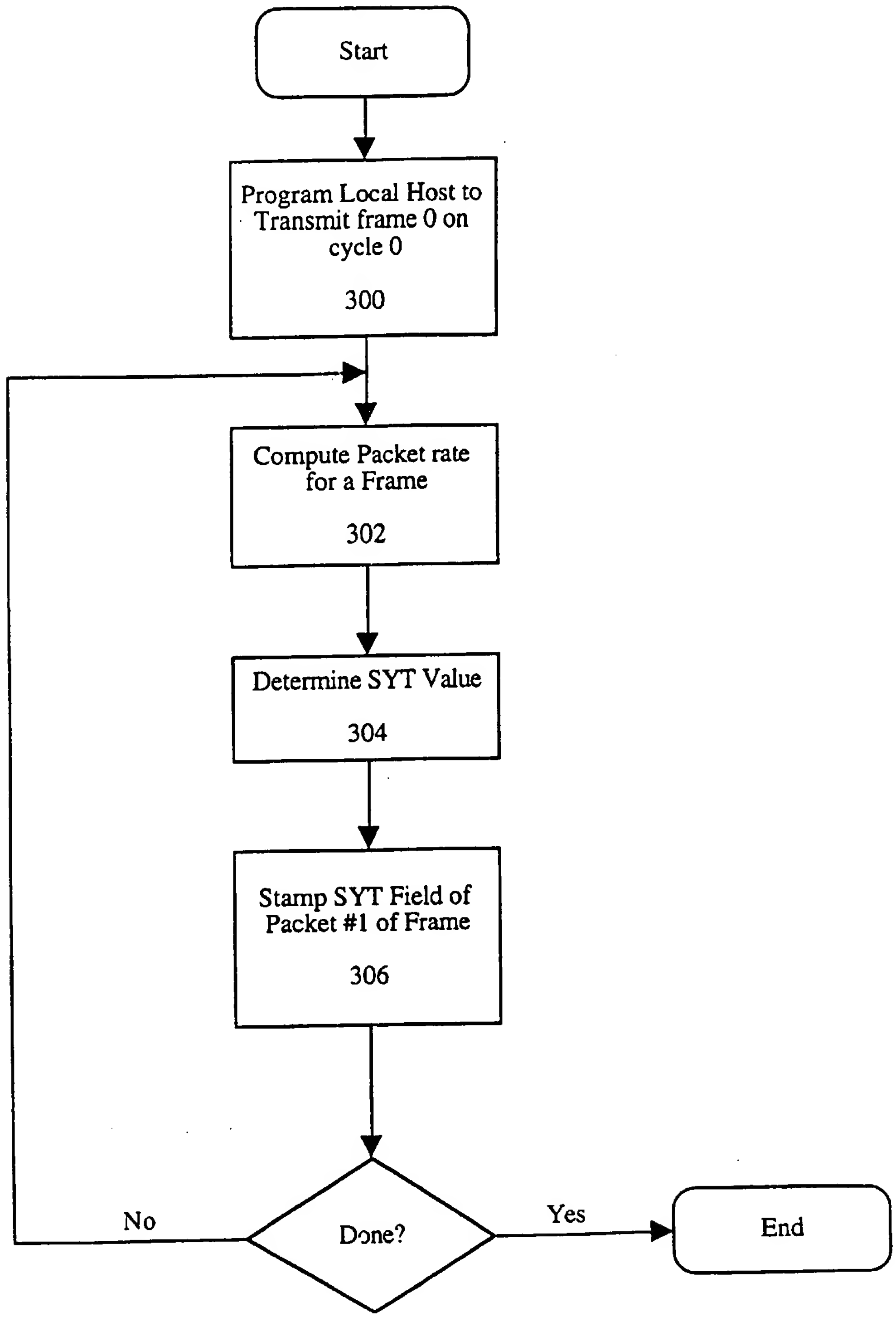

Start
Program Local Host to Transmit frame 0 on cycle 0
300
Compute Packet rate for a Frame
302
Determine SYT Value
304
Stamp SYT Field of Packet #1 of Frame
306
Done?
No
Yes
End

Fig. 6

METHOD FOR SETTING TIME STAMP IN SYT FIELD OF PACKET HEADERS FOR IEEE-1394 DEVICES

FIELD OF THE INVENTION

This invention relates generally to data communications and, more particularly, to a method for controlling isochronous data communications within a digital system having a bus architecture that complies with the IEEE-1394 Standard for a High Performance Serial Bus.

BACKGROUND

The components of a computer or other digital system are typically coupled to a common bus for communicating information to one another. Various bus architectures are known in the prior art, and each bus architecture operates according to a communications protocol that defines the manner in which data transfer between components is accomplished.

The Institute of Electrical and Electronic Engineers (IEEE) has promulgated a number of different bus architecture standards including IEEE standards document 1394, entitled *Standard for a High Performance Serial Bus* (hereinafter "IEEE-1394 Serial Bus Standard"). A typical serial bus having the IEEE-1394 standard architecture is comprised of a multiplicity of nodes that are interconnected via point-to-point links, such as cables, that each connect a single node of the serial bus to another node of the serial bus. Data packets are propagated throughout the serial bus using a number of point-to-point transactions, wherein a node that receives a packet from another node via a first point-to-point link retransmits the received packet via other point-to-point links. A tree network configuration and associated packet handling protocol ensures that each node receives every packet once. The serial bus of the IEEE-1394 Serial Bus Standard may be used as an alternate bus for the parallel backplane of a computer system, as a low cost peripheral bus, or as a bus bridge between architecturally compatible buses.

A communications protocol of the IEEE-1394 Serial Bus Standard specifies two primary types of bus access: asynchronous access and isochronous access. Asynchronous access may be either "fair" or "cycle master". Cycle master access is used by nodes that need the next available opportunity to transfer data. Isochronous access is used by nodes that require guaranteed bandwidth, for example, nodes transmitting video or audio data. The transactions for each type of bus access are comprised of at least one "subaction", wherein a subaction is a complete one-way transfer operation.

In the case of, for example, digital video data transfers within digital systems conforming to the IEEE-1394 Serial Bus Standard, the video data may be transferred for example, between a mass storage device (e.g., a digital memory such as a hard disk drive, a flash memory device or other storage medium) and a digital video camera or other recorder (e.g., to store an edited video sequence) under the control of a computer processor or other device (e.g., a DMA controller). The video data is transferred as a series of frames, each frame being made up of a number of data packets. The individual data packets include a number header fields (which include various information regarding the data as well as addressing information) as well as the video data itself.

In order to ensure that each frame of the video data is played out in the proper sequence, the frames must be "time

stamped" with an appropriate frame presentation time (e.g., measured in terms of "cycle time" of an isochronous transaction on a bus complying with the IEEE-1394 Serial Bus Standard) when they are recorded. The frame presentation time for individual frames of data is recorded in a particular header field, referred to as an SYT field, of the first packet of each frame (note that for non-video applications, the concept of a "frame" is not used and the SYT field may be located and stamped in each packet or only some of the packets of a data transfer). In essence, the frame presentation time "stamped" in the SYT field of the packet header is an indication to the receiver of the time that the frame should be played out. For digital video data, the frame presentation time may be up to 450 μsec. in the future. That is, from the point of view of the receiver, the SYT field frame presentation stamp value for a given frame of data must be within 450 μsec. of the time the first packet in that frame is received. Thus, in the example given above, when the digital video data is transferred from the mass storage device to the recording medium, the computer processor or other device which is controlling the transfer must insert appropriate frame presentation time stamp (or SYT) values into the SYT fields of the first packet in each frame of the video data. Note that the 450 μsec. requirement is specific to video data and other types of data, e.g., MIDI audio data, may have other frame presentation time requirements.

In the past, such time stamping operations have required the use hardware interrupt procedures to determine a current cycle time which could then be written to the SYT field of a packet. However, there are times at which such interrupt procedures cannot be completed within the 450 μsec. (e.g., for digital video applications) time limitation. As a result, some frames of data are "lost" and any resulting display of the entire video data stream is degraded. It would therefore be desirable to have other solutions which do not rely on the hardware interrupt procedures of the past for time stamping the SYT fields of data in a digital network complying with the IEEE-1394 Serial Bus Standard.

SUMMARY OF THE INVENTION

Methods for controlling isochronous data communications within a digital system having a bus architecture that complies with the IEEE-1394 Standard for a High Performance Serial Bus are described.

In one embodiment, a presentation time stamp field of a packet of data for transmission in a digital network is set with a presentation time value determined according to a computed packet rate for the data.

In a further embodiment, a presentation time stamp field of a first packet of a second frame of data for transmission in a digital network is set with a presentation time value determined according to a computed packet rate for the data. The packet rate may be computed by measuring a difference between a desired presentation time value of a first packet in a first frame of the data and an actual transmission time of the first packet of the first frame. The first frame preceding the second frame in time of transmission within the network.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention is illustrated by way of example and not limitation in the figures of the accompanying drawings, in which like references indicate similar elements and in which:

FIG. 1 illustrates a digital system having a serial bus made up of a number of nodes and supporting the control of isochronous data according to one embodiment of the present invention;

FIG. 2 shows an exemplary software architecture supporting a Data Stream Command Language (DCL) according to one embodiment of the present invention;

FIG. 3 illustrates a nil terminated DCL program having DCLJump commands to allow for the transmission of varying packet rates within a digital network according to one embodiment of the present invention;

FIG. 4 illustrates one example of a DCL program which may utilize a ModifyDCLJump operation to control a data stream in a digital system according to one embodiment of the present invention;

FIG. 5 is a flow diagram illustrating one embodiment of the present invention and, particularly, a process for calculating a frame presentation time value from a computed data packet rate and the determination of a new packet rate based thereon; and

FIG. 6 is a flow diagram which summarizes a process for determining a frame presentation time value for frames of data to be transmitted in a digital network according to one embodiment of the present invention.

DETAILED DESCRIPTION

As described herein, methods for controlling isochronous data communications within a digital system having a bus architecture that complies with the IEEE-1394 Standard for a High Performance Serial Bus are provided. For example, FIG. 1 shows an exemplary digital system utilizing the methods of the present invention. As will be described in detail below, in one embodiment presentation time stamp fields of data packets for transmission in the digital network may be set with a presentation time value determined according to a computed packet rate for the data. The packet rate may be computed by measuring a difference between a desired presentation time value and an actual transmission time of the first packet of a frame of data transmitted prior to a current frame of interest. Although the present invention is described with reference to the transmission of video data within digital network **5**, it should be noted that for non-video applications the concept of a "frame" is not used and the SYT field may be located and stamped in each packet or only some of the packets of a data transfer. The present invention is applicable to any digital data transmission within a network having a bus architecture that complies with the IEEE-1394 Standard for a High Performance Serial Bus and is not limited to video data transfer applications.

Some portions of the detailed description which follows are presented in terms of data structures, algorithms and symbolic representations of operations on data within a computer network and/or a computer memory. These descriptions and representations are the means used by those skilled in the computer science arts to most effectively convey the substance of their work to others skilled in the art. An algorithm is here, and generally, conceived to be a self-consistent sequence of steps leading to a desired result. The steps are those requiring physical manipulations of physical quantities. Usually, though not necessarily, these quantities take the form of electrical and/or magnetic signals capable of being stored, transferred, combined, compared and otherwise manipulated. It has proven convenient at times, principally for reasons of common usage, to refer to these signals as bits, values, elements, symbols, characters, terms, numbers or the like. It should be borne in mind, however, that all of these and similar terms are to be associated with the appropriate physical quantities and are merely convenient labels applied to these quantities. Unless specifically stated otherwise, it will be appreciated that throughout the description of the present invention, use of terms such as "processing", "computing", "calculating", "determining", "displaying", or the like, refer to the actions and processes of a computer or other digital system that manipulates and transforms data represented as physical (electronic) quantities within the computer system's registers and memories into other data similarly represented as physical quantities within the computer system memories or registers or other such information storage, transmission or display devices.

The digital system **5** of FIG. **1** includes a central processing unit (CPU) **10**, a monitor **18**, a printer **26**, a video camera **32**, a video cassette recorder (VCR) **36**, a keyboard **42**, and a mouse **46**. The CPU **10** includes an internal hard drive **14** and a memory (not shown). Each of the devices of digital system **5** is coupled to a local node of the serial bus. In general, the device to which a node is coupled acts as the "local host" for that node. For example, the CPU **10** is the local host for the CPU node **12**; the monitor **18** is the local host for the monitor node **16**; the printer **26** is the local host for the printer node **24**; the video camera **32** is the local host for the video camera node **30**; the VCR **36** is the local host for the VCR node **34**; the keyboard **42** is the local host for the keyboard node **40**; the mouse **46** is the local host for the mouse node **44**; and the internal hard drive **14** is the local host for the internal hard drive node **15**. Those skilled in the art will appreciate that it is not always necessary for every node to have a local host, nor is it necessary that a local host always be powered.

A point-to-point link such as cable **20** is used to connect two nodes to one another. CPU node **12** is coupled to internal hard drive node **15** by an internal link **21**, to monitor node **16** by cable **20**, and to keyboard node **40** by a cable **20***e*. The keyboard node **40** is coupled to the mouse node **44** by a cable **20***f*. The monitor node **16** is coupled to the nodes of the other peripherals (not shown) by cable **20***a* and to the printer node **24** by cable **20***b*. The printer node **24** is coupled to the video camera node **30** by cable **20***c* and to the VCR node **34** by cable **20***d*. Each of the cables **20**-**20***f* and the internal link **21** may be constructed in accordance with the IEEE-1394 Serial Bus Standard and may include a first differential signal pair for conducting a first signal, a second differential signal pair for conducting a second signal, and a pair of power lines.

Each of the nodes **12**, **15**, **16**, **24**, **32**, **34**, **40** and **44** may have identical construction, although some of the nodes, such as mouse node **44**, can be simplified because of their specific functions. Thus, the nodes can be modified to meet the needs of a particular local host. For example, each node may have one or more ports, the number of which is dependent upon its needs. For example, CPU node **12**, as illustrated, has 3 ports, while the mouse node **44** has only 1 port.

Digital system **5** supports the transfer of data packets (e.g., made up of digital video and/or audio) associated with a data stream. For example, digital video data from hard drive **14** may be transmitted to video camera **32**, e.g., for recording onto digital video tape. The video data transmitted to video camera **32** will comprise isochronous data packets in accordance with the IEEE-1394 Serial Bus Standard. Each of these isochronous data packets will include header information and payload information. The payload information comprises the video data to be recorded. The header information is used for routing the video data to the video camera **32** and for error detection and correction. In addition, and in accordance with one embodiment of the present invention, the header information of the first packet for each frame of the video data includes a presentation time

stamp value within an SYT field of the header. The presentation time stamp value is determined according to a computed packet rate for the data, as discussed further below.

The video data is transmitted on a particular isochronous channel within digital system 5. The isochronous channel is identified by a channel identification number (channel ID). The channel ID is maintained in a data record stored in the digital system 5 (e.g., in the memory associated with CPU 10) and is used by the various application programs and driver routines running on CPU 10 to coordinate the transfer of data. The use of a common channel ID allows the interoperation of application programs, driver routines, and other software routines which otherwise may not be capable of operating together.

When video data is to be transmitted, the present invention takes advantage of a feature of currently available hosts designed for use with the IEEE-1394 Serial Bus Standard. Current hosts (and/or their associated nodes) may be programmed to begin transmitting isochronous data on a particular cycle number. The cycle number is determined from the cycle time found in the cycle start packet broadcast by the cycle master on the bus. The cycle time indicates the cycle number. Thus, according to the present invention, the local host and/or its respective node, associated with the data transmitting device is programmed to begin transmission on a particular isochronous cycle number ("N"). Then, assuming that "M" data packets are to be transmitted for each frame (i.e., M is a computed packet rate for the data) and that "x" represents the frame number, then the frame presentation time value to be stamped in the SYT field of the first packet for each frame of data to be transmitted is given by:

$$SYT[x]=N+Mx+a, \qquad (1)$$

where the value "a" is a precomputed offset. In the case of digital video,

$$0<a<450\ \mu\text{sec.}$$

For example, if the transmitter is programmed to begin transmitting frame O on cycle O (i.e., N 0), then

$$SYT[0]=a$$

$$SYT[1]=M+a$$

$$SYT[2]=2M+a$$

.

.

.

$$SYT[y]=yM+a$$

This SYT value may be precomputed for the data (e.g., using equation (1) above) in advance of the actual data transmission, thus avoiding the hardware interrupt latency issues described above. When the packets are ready for transmission, the appropriate SYT value is written into the SYT field of the first packet (e.g., using conventional techniques for establishing the header of a packet to be transmitted on a bus complying with the IEEE-1394 Serial Bus Standard) for each corresponding frame of data to be transmitted.

Unfortunately, M, the packet rate for the data, may not be an integer value. For example, in the case of digital video to be displayed on an NTSC compatible device (e.g., a television or video monitor), M≈266.973. This presents a problem because only an integer number of packets can be transmitted for each frame (i.e., there are no partial packets and even an empty packet counts as one). In addition, there are occasions where no packets are transmitted during an isochronous cycle (e.g., a "missed" cycle). Each of these situations must be accounted for.

The present invention accommodates the need for a non-integer M as follows. Assume that the desired average M (hereafter M_{av}) to maintain proper synchronization is 266.5 (as indicated above, NTSC compatibility requires M≈266.973, however, for simplicity in the following examples, 266.5 is used). To achieve an overall M_{av}=266.5, sometimes the transmitter will need to send 266 packets/frame and sometimes 267 packets/frame. To accommodate such varying frame lengths, a data stream command language is provided. The data stream command language (DCL) is, in one embodiment, a set of commands that control data flow into or out of a data stream, such as the data stream between hard drive 14 and camera 32 discussed above. A collection of DCL commands are connected together into a linked list to form a DCL program which can be assigned to a particular data stream such as a data stream associated with an isochronous channel. The default execution order of a DCL program is to start with the first DCL command in the program and to follow the DCL command links. This execution order may be changed by using DCL jump commands. Accordingly, if every frame in a data stream is characterized by a DCL program which includes 267 packets/frame, appropriate DCL jump commands may be inserted into the program stream to allow a transmitter to send 267 packets per frame or to send only 266 packets per frame.

FIG. 2 illustrates a software architecture 48 supporting the DCL. An application program 50 running on digital system 5 (e.g., in conjunction with an operating system 52) may provide a user interface which allows a user to control the transfer of digital video data to camera 32. A driver 54 called by the application program 50 may utilize DCL services provided by a services library 56 to develop a DCL program to control the data streams which make up the transfer. Each data stream contains data packets as described above and in the IEEE-1394 Serial Bus Standard.

The DCL program generated by the driver 54 will consist of a nil terminated linked list of DCL commands. At least a minimum set of commands are provided to control the data stream. For example, a DCLSendPacketStartOp command may be used to specify the first part of a packet to be sent to a data stream (e.g., from hard drive 14). Subsequent parts of a packet may be specified using a DCLSendPacketOp command. A packet is defined as a contiguous string of DCL packet commands that start with a DCL packet start command and end with any DCL command that is not a DCL packet command. Thus, scatter/gather lists may be used in constructing packets. To determine the total size of a packet, a DCL compiler (e.g., an interface module 58) may sum respective size fields in any DCL packet start and packet commands defining the packet of interest. DCL send packet buffers need not include a packet header. Instead, a packet header will be constructed by the compiler, based upon the channel number for the data stream associated with the DCL program, any tag and sync bits specified by a DCLSetPacketAttributes command and the computed length of the packet. Exemplary DCLCommands and records, e.g., DCLSendPacketStartOp, DCLSendPacketOp, DCLTransferPacket, DCLReceivePacketStartOp, DCLReceivePacketOp, DCLReceiveBufferOp, DCLCallProcDCLSetPacketAttributesOp, DCLLabelOp,

and DCLJumpOp are described in detail in U.S. patent application Ser. No. 08/731,173, entitled "Software Architecture for Controlling Data Streams", filed Oct. 10, 1996, by Erik P. Staats and assigned to the Assignee of the present application, the entire disclosure of which is incorporated herein by reference.

The DCLJumpOp command is used to change the default order of a DCL program. For example, the DCLJumpOp command may be used by a driver to create a looping DCL program, as shown in FIG. 3. The DCL program includes the DCLJump command which allows for the looping (e.g., by pointing to an earlier DCL label in the program stream) and is also nil terminated to allow for ease of compilation.

The DCLJumpOp command may use the following record.

```
struct DCLJumpStruct
{
      DCLCommandPtr          pNextDCLCommand;
      UInt32                 compilerData;
      UInt32                 opcode;
      DCLLabelPtr            pJumpDCLLabel;
};
typedef struct DCLJumpStruct DCLJump
                             *DCLJumpPtr;
      pNextDCLCommand        Link to next DCL command in
                             program.
      compilerData           DCL compiler's private data.
      opcode                 Opcode specifying type of DCL
                             command.
      pJumpDCLLabel          Pointer to DCL label to jump to.
```

The pJumpDCLLabel field specifies the DCL label command to jump to.

The services library **56** also provides a means to update a DCL program. The SetDCLProgramCompilerNotificationProc sets the routine to call when a DCL program is updated and is typically called by an interface module. If a driver wishes to change a DCL command while a DCL program is running, the compiler must be notified to change the corresponding DMA commands. This may happen, for example, if a driver wishes to change the destination of a DCL jump command by calling DCLModifyJump (see below). The DCLModifyJump routine will call the DCL program's notification routine and pass it the DCL jump command that has been modified. The modification routine then makes any changes necessary to change the target of the jump command. Typically, this will involve changing the destination of a DMA branch command. A sample call is as follows.

```
OSStatus      SetDCLProgramCompilerNotificationProc (
      DCLProgramID                     dclProgramID,
      DCLCompilerNotificationProc      dclCompilerNotificationProc);
-->   dclProgramID                     DCLProgramID to set start
                                       event of.
-->   dclCompilerNotificationProc      Proc to call on DCL program
                                       updates.
```

The ModifyDCLJump operation is provided to modify a DCL jump command in a DCL program that is currently running. A sample call for this routine is as follows.

```
OSStatus      ModifyDCLJump         (
      DCLProgramID                  dclProgramID,
      DCLJumpPtr                    pDCLJump,
      DCLLabelPtr                   pDCLLabel);
-->   dclProgramID                  DCLProgramID to set start
                                    event of.
-->   pDCLJump                      Pointer to DCL jump
                                    command to modify.
-->   pDCLLabel                     Pointer to new destination of
                                    the DCL jump command.
```

This routine may be called while a DCL program is in progress as illustrated with reference to FIG. 4. Suppose the DCL program **100** shown in FIG. 4 has been created to transfer video data to camera **32**, e.g., for recording on a video tape. Ideally, as a frame of data is transmitted from a respective buffer (e.g., in memory or hard drive **14**) to the camera **32** as part of an associated data stream, the buffer's contents will be updated by the associated driver with a new frame of data. However, to account for situations where the packet rate must be varied (e.g., to achieve a non-integer average packet rate), the program is written to always transmit a certain number of packets per frame (e.g., 267 packets/frame) unless certain jump instructions (e.g., jump instruction **100**) are modified as the program is running.

To further illustrate, as shown in FIG. 4, a driver has created a frame transmission operation having a number of DCL send packet commands **104**. In total, **267** packets/frame are transmitted. At the end of each frame transmission operation is a DCL call procedure (call proc) command **106** which notifies the driver that a frame of data has been sent and that the associated buffer should be updated with new data. After the send packet command **104** regarding packet number **266**, the driver has placed a DCL jump command **102**. So long as the data in a buffer following a DCL jump command should be transmitted, the DCL jump command will simply jump to a DCL label (note, for clarity the label commands have not been shown in FIG. 4) before that buffer, allowing the associated data in the buffer to be passed to the data stream. However, if the buffer following a DCL jump command contains a packet which should not be transmitted (e.g., when only 266 packets/frame are to be transmitted), the DCL jump command will jump to a DCL label beyond that packet. Thus, whenever the driver is notified to only transmit 266 packets, it will call ModifyDCLJump to set the DCL jump command before the 267th packet to point to a DCL label beyond the associated buffer (i.e., DCL jump command **102** will be modified to point to DCL Call Proc command **106**). thus, the 267th packet (which, is an empty packet as there are typically fewer than 266 packets of true video data per frame—the remaining packets being empty packets) will be sent.

To determine when the driver should be notified to only send 266 packets, the following routine is used. A value "Δ" is computed so that when

$$\Delta \geq 1, M=267, \text{ and}$$

$$\Delta < 1, M=266, \text{ where}$$

$$\Delta_z = SYT[z] - (\text{cycle start \# for frame z}) - a.$$

The cycle start # for the frames is always incremented by the current M to achieve the desired overall average frame rate (M_{av}=266.5 in this example). The current M is then adjusted (e.g., using the ModifyDCLJump procedure described above) according to the Δ for the previously transmitted frame.

To illustrate, assume again that the desired M_{av}=266.5 and that the host transmitter is programmed to begin transmitting frame **0** on cycle **0**. If M is established to begin at 266 and a=2 (i.e., representing an offset of twice the frame period or 250 μsec.), then new values for M are computed as shown in Table 1 below:

TABLE 1

M	SYT[x] = N + M_{av}x + a	cycle #to begin transmission of frame x	Δ
266	SYT[0] = 0 + (266.5)(0) + 2 = 2	frame[0] begins on cycle 0	0
266	SYT[1] = 0 + (266.5)(1) + 2 = 268.5	frame[1] begins on cycle 266	0.5
266	SYT[2] = 0 + (266.5)(2) + 2 = 535	frame[2] begins on cycle 532	1.0
267	SYT[3] = 0 + (266.5)(3) + 2 = 801.5	frame[3] begins on cycle 799	0.5
266	SYT[4] = 0 + (266.5)(4) + 2 = 1068	frame[4] begins on cycle 1065	1.0
267	SYT[5 = 0 + (266.5)(5) + 2 = 1334.5	frame[5] begins on cycle 1332	0.5
.	.	.	.
.	.	.	.
.	.	.	.

From the above Table 1, it can be seen that an SYT value for a current frame is computed using a current M. Then, Δ is computed for the current frame and M is adjusted accordingly for the next frame to achieve an M_{av} of 266.5.

In addition, the present invention (at least in this one embodiment) involves setting a presentation time stamp field (the SYT field) of a first packet of a given frame of data for transmission in a digital network with a presentation time value (e.g., the SYT value from column 2 of the above table) determined according to a computed packet rate (e.g., M) for the data. The packet rate M for any given frame may be computed as the difference between the SYT value for the previous frame and that previous frame's corresponding cycle # (i.e., the cycle # at which the first packet of that previous frame was transmitted) minus a (where a=a pre-computed offset). This method is summarized in FIG. **5**.

As shown, at step **200**, a current packet rate (M) is used to compute a current SYT value for a frame as described above. Then, in step **202**, a new Δ is determined as the difference between the current SYT value and the corresponding cycle # for the frame minus a. Based on the new Δ, a new packet rate is determined at step **204**. This new packet rate may be used to compute the next M value for the next frame, and so on until all the packets have been transmitted.

If no missed cycles had to be accounted for (i.e., if it were assured that a packet were transmitted on each cycle), the above scheme would be complete. However, sometimes a packet is not transmitted every cycle, e.g., if the transmitter is denied access to the bus on a given cycle. In this event, the "missed" packet will be transmitted in the next cycle, i.e., one cycle late. The above algorithm assumes that one packet was transmitted every cycle and would not account for the missed cycle. For example, if in the above table frame[**2**] actually began transmission on cycle #**533** and not **532** as predicted, the actual Δ for frame[**2**] would be 0 and not 1.0. Thus, at this point M should be held at 266 to accommodate the missed cycle. Instead, however, the above algorithm predicted that Δ for frame[**2**] was 1.0 and, accordingly, changed M to 267. This causes video to be lost as shown in Table 2 below:

TABLE 2

M (based on predicted Δ)	SYT[x]	predicted cycle #to begin transmission of frame x	Δ (predicted)	actual cycle # transmission of frame x began	Δ (actual)
266	SYT[O] =2	frame[0] →0	0	0	2
266	SYT[1] = 268.5	frame[1] → 266	0.5	266	0.5
266	SYT[2] = 535	frame[2] → 532	1.0	533*	0**
267	SYT[3] = 801.5	frame[3] → 799	0.5	800	-0.5***

*indicates missed cycle
**M should be held at 266 at this point to accommodate the missed cycle
***indicates lost data because the frame was sent too late

To accommodate the possibility that missed cycles may occur, the actual cycle # that a frame begins transmission on must be determined and accounted for. Current local hosts and/or their associated nodes developed to be compatible with the IEEE-1394 Serial Bus Standard include a register which maintains the current cycle time. This register can be read during transmit operations and, as a result, a track of the actual cycle time (or cycle #) that transmission of a frame is commenced may be determined. This actual frame transmission time may be compared with (or used in lieu of) the predicted transmission time to compute an appropriate M so as to maintain frame synchronization.

Thus a scheme for controlling isochronous data communications within a digital system having a bus architecture that complies with the IEEE-1394 Standard for a High Performance Serial Bus has been described. The process is summarized in FIG. **6** as follows. At step **300**, a host is programmed to begin transmission of data on a desired cycle #, e.g., cycle **0**. The packet rate for the data is computed (or preset for the first frame as discussed above) at step **302** and, at step **304**, the appropriate SYT value based on the computed packet rate is determined. This SYT value may be stamped into the SYT field of the first packet of the frame at step **306**. The process of determining a packet rate (e.g., based on the SYT value of a previous frame as discussed above) continues until all frames of the data have been transmitted or the process has otherwise been terminated. In

the foregoing specification, the invention has been described with reference to specific exemplary embodiments thereof. It will, however, be appreciated by those skilled in the art that various modifications and changes may be made thereto without departing from the broader spirit and scope of the invention as set forth in the appended claims. The specification and drawings are accordingly, to be regarded in an illustrative rather than a restrictive sense.

What is claimed is:

1. A computer readable medium having stored thereon a plurality of sequences of instructions, said plurality of sequences of instructions including sequences of instructions which, when executed in a digital network including a plurality of nodes interconnected by a plurality of point-to-point links, cause one of said nodes to set a presentation time stamp field of a first packet of a second frame of data for transmission in the digital network with a presentation time value determined according to a computed packet rate for the data, wherein the data for transmission includes a plurality of frames and wherein each one of the plurality of frames includes a plurality of packets, and cause the one of said nodes to set an actual cycle time of the packet of data and wherein each one of the plurality of frames includes a variable length.

2. A computer readable medium having stored thereon a plurality of sequences of instructions, said plurality of sequences of instructions including sequences of instructions which, when executed in a digital network including a plurality of nodes interconnected by a plurality of point-to-point links, cause one of said nodes to set a presentation time stamp field of a first packet of a second frame of data for transmission in the digital network with a presentation time value determined according to a computed packet rate for the data, wherein the data for transmission includes a plurality of frames and wherein each one of the plurality of frames includes a plurality of packets, and cause the one of said nodes to set an actual cycle time of the packet of data and cause said node to compute the packet rate by measuring a difference between a desired presentation time value of a first packet of a preceding frame of the data and an actual transmission time of the first packet of the preceding frame and cause said node to compute a presentation time stamp field of the first packet of the first frame equal to N+MX+A where N is equal to a cycle number of the first packet of the first frame, M is equal to a computed packet rate, X is equal to a frame number corresponding to the first frame, and A is equal to a precomputed offset.

3. A method comprising setting a presentation time stamp field of a packet of data for transmission in a digital network with a presentation time value determined according to a computed packet rate for the data, and setting an actual cycle time of the packet of data wherein said packet of data is a first packet of a second frame of data for transmission in said digital network, wherein the data for transmission includes a plurality of frames, wherein each one of the plurality of frames includes a plurality of packets, wherein the packet rate is computed by measuring a difference between a desired presentation time value of a first packet of a first frame of the data and an actual transmission time of the first packet of the first frame, the first frame preceding the second frame in time of transmission within the network, and setting a presentation time stamp field of the first packet of the first frame to a value equal to N+MX+A where N is equal to a cycle number of the first packet of the first frame, M is equal to a computed packet rate, X is equal to a frame number corresponding to the first frame, and A is equal to a precomputed offset.

4. A method comprising:

setting a presentation time stamp field of a first packet of data of a second frame for transmission in a digital network with a presentation time value determined according to a computed packet rate for the data; and

setting an actual cycle time of the packet of data;

wherein the data for transmission includes a plurality of frames and wherein each one of the plurality of frames includes a plurality of packets;

wherein the packet rate is computed by measuring a difference between a desired presentation time value of a first packet of a first frame of the data and an actual transmission time of the first packet of the first frame, the first frame preceding the second frame in time of transmission within the network, and setting a presentation time stamp field of the first packet of the first frame to a value equal to N+MX+A where N is equal to a cycle number of the first packet of the first frame, M is equal to a computed packet rate, X is equal to a frame number corresponding to the first frame, and A is equal to a precomputed offset.

5. A computer readable medium having stored thereon a plurality of sequences of instructions, said plurality of sequences of instructions including sequences of instructions which, when executed in a digital network including a plurality of nodes interconnected by a plurality of point-to-point links, cause one of said nodes to:

set a presentation time stamp field of a first packet of a second frame of data for transmission in the digital network with a presentation time value determined according to a computed packet rate for the data, wherein the data for transmission includes a plurality of frames and wherein each one of the plurality of frames includes a plurality of packets;

set an actual cycle time of the packet of data;

compute the packet rate by measuring the difference between a desired presentation time value of a first packet of a proceeding frame of the data and an actual transmission time of the first packet of the proceeding frame; and

compute a presentation time stamp field of the first packet of the first frame equal to N+MX+A where N is equal to a cycle number of the first packet of the first frame, M is equal to a computed packet rate, x is equal to a frame number corresponding to the first frame, and A is equal to a precomputed offset.

* * * * *

#2 Am

OIPE JCTS
OCT 2 8 2004
PATENT & TRADEMARK OFFICE

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/249,642	02/12/1999	QUAN A. VU	SONY-11300	1161

28960 7590 06/04/2004

HAVERSTOCK & OWENS LLP
162 NORTH WOLFE ROAD
SUNNYVALE, CA 94086

EXAMINER	
WILSON, JACQUELINE B	

ART UNIT	PAPER NUMBER
2612	32

DATE MAILED: 06/04/2004

DOCKETED

Please find below and/or attached an Office communication concerning this application or proceeding.

JUN 07 2004

Office Action Summary	Application No.	Applicant(s)	
	09/249,642	VU ET AL.	
	Examiner	**Art Unit**	
	Jacqueline Wilson	2612	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *03 May 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☐ Claim(s) *1,2,4-8 and 10-32* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☐ Claim(s) *1,2,4-8 and 10-32* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other: _____.

DETAILED ACTION

1. Applicant's arguments filed 05/03/04 have been fully considered but they are not persuasive. The applicant continues to argue that the prior art, Staats, fails to teach evenly distributing the x number of first data blocks among the y number of second data blocks. Please refer to the arguments in paper numbers 26 and 28. As for the applicants argument indicating that there is no hint, teaching or suggestion to even warrant an obviousness determination and to do so would be impermissibly use hindsight to make a rejection based on obviousness. Hindsight reasoning is inapplicable to this application and only refers to 35 USC § 103. The examiner's rejection is based solely on *35 USC § 102* in which the rejections are maintained below. As for the newly added limitation, Staats teaches calculating a ratio by determining when to insert 266 packets/frame in the data stream of 267 packets/frame. Therefore, the rejection is maintained.

Claim Rejections - 35 U.S.C. § 102

1. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless --
>
> (e) the invention was described in a patent granted on an application for patent by another filed in the United States before the invention thereof by the applicant for patent, or on an international application by another who has fulfilled the requirements of paragraphs (1), (2), and (4) of section 371(c) of this title before the invention thereof by the applicant for patent.

2. Claims 1, 2, 4-8, 10-20, and 23-25 are rejected under 35 U.S.C. 102(e) as being anticipated by Staats (US 6,373,821).

Regarding Claim 1, Staats'821 teaches transmitting information from a source device at a predetermined rate comprising forming x number of first data blocks wherein each of the first data blocks contains n units of data (267 packets/frame; col. 6, lines 7+), and forming y number of second data blocks wherein each of the second data blocks contains m units of data (266 packets/frame) wherein m is not equal to n. Staats'821 further teaches that each data stream contains these data packets in which 267 packets/frame of data is transmitted and sometimes 266 are need to be transmitted. This inherently teaches combining x number of first data blocks and y number of second data blocks into a data stream to achieve the predetermined rate, wherein the first data blocks and the second data blocks are of a same type and have the same characteristics (video data). As for the limitation of the x number of first data blocks are evenly distributed among the y number of second data blocks, the examiner believes Staats teaches this concept. In order to produce an IEEE-1394 serial bus standard, Staats teaches that the NTSC compatibility requires the data stream to equal 266.973, as discussed above. In order to achieve this data rate, uniformity in the data stream is inherent in the system of Staats. Staats discloses that after a certain number of x data blocks (267) are present in the data stream, a jump command includes the y data block (266) into the stream. Therefore, to maintain a proper stream, uniformity of the data blocks must be present. Since the data stream is not restricted to a time period, over time the data stream will eventually repeat itself, thereby producing an

evenly distributed x and y data blocks having first and second frames forming a repeating pattern within the data stream. Table 1 explains calculating a ratio of first data blocks to second data blocks to achieve the predetermined rate.

The applicant argues that the prior art fails to teach forming x number of first data blocks each containing n units of data, forming y number of second data blocks each containing m units of data and combining x number of first data blocks and y number of second data blocks into a data stream to achieve the predetermined rate and evenly distributing the x number data blocks among the y number of data blocks. Closely reviewing the Staats reference, the examiner still believes that the prior art teaches the applicants claimed limitations. Staats teaches in Table 1 equations used in determining when to transmit data blocks. Although Staats used 266.5 for example purposes, the examiner uses 266.973 (which is closest to 267) as discussed in column 6, lines 10+. Beginning in cycle 0, the data is given below:

		Cycles begins	**Δ**
267	(266.973)(0) + 2 = 2	0	0
267	(266.973)(1) + 2 = 268.973	267	.027
267	(266.973)(2) + 2 = 535.946	534	.054
	.		
	.		
	.		
	(266.973)(10) + 2 = 2671.73	2670	.27
	.		
	.		
	.		
267	(266.973)(35) + 2 = 9346.055	9345	.945
267	(266.973)(36) + 2 = 9613.028	9612	.972
267	(266.973)(37) + 2 = 9880.001	9879	.999
267	(266.973)(38) + 2 = 10146.974	10146	1.026
266	(266.973)(39) + 2 = 10413.947	10412	.053
	.		

.

.

267	(266.973)(75) + 2 = 20024.975	20024	1.025
266	(266.973)(76) + 2 = 20297.948	20290	.052

Staats uses 266.5 for convenience in showing when to include 266 and 267 data packets in the data stream. However, the examiner uses the targeted value 266.973. In this case, after calculating the first two values, the cycle repeats every 37th packet. As shown above when x=2, 39, 76, etc, the DCL jump command will include packet 266 and will repeat over time (see col. 8-col. 9). This reads on the limitation of calculating a ratio of first data blocks to second data blocks to achieve the predetermined rate (37:1) and evenly distributing x number of first data blocks among the y number of second data blocks thereby forming a repeating pattern of the first data blocks and second data blocks within the data stream.

Regarding Claim 2, Staats'821 teaches transmitting the data stream from the source device at the predetermined rate (col. 10, lines 57+ teaches the host is programmed to begin transmission of data at a desired cycle).

Regarding Claim 4, Staats'821 teaches digital video data (col. 3, lines 30-33).

Regarding Claim 5, Staats'821 teaches n, m, x, and y are integer values (x and y are each frame, and n and m are 266 and 267).

Claim 6 is analyzed and discussed with respect to Claim 1 (source and receiving devices are the host computer and camera).

Claim 7 is analyzed and discussed with respect to Claim 5. (See rejection of Claim 5 above.)

Claim 8 is analyzed and discussed with respect to Claim 2 with the further limitation of the data stream conforming to the standards of an IEEE 1394-1995 network (col. 3, lines 24+).

Claim 10 is analyzed and discussed with respect to Claim 8. (See rejection of Claim 8 above.)

Regarding Claim 11, Staats'821 teaches the source and receiving device are coupled together within a network (see fig. 1).

Claim 12 is analyzed and discussed with respect to Claim 8. (See rejection of Claim 8 above.)

Claim 13 is analyzed and discussed with respect to Claim 1. (See rejection of Claim 1 above.)

Claim 14 is analyzed and discussed with respect to Claim 5. (See rejection of Claim 5 above.)

Regarding Claim 15, Staats'821 teaches an interface coupled to the controller and configured for connecting to a network (fig. 1, 12).

Claim 16 is analyzed and discussed with respect to Claim 8. (See rejection of Claim 8 above.)

Claim 17 is analyzed and discussed with respect to Claim 1. (See rejection of Claim 1 above.)

Claim 18 is analyzed and discussed with respect to Claim 5. (See rejection of Claim 5 above.)

Claim 19 is analyzed and discussed with respect to Claim 6 (see also col. 8, lines 15-16). (See rejection of Claim 6 above.)

Claim 20 is analyzed and discussed with respect to Claims 6 and 19 . (See rejection of Claims 6 and 19 above.)

Claim 23 is analyzed and discussed with respect to Claim 8. (See rejection of Claim 8 above.)

Claim 24 is analyzed and discussed with respect to Claims 6 and 11. (See rejection of Claims 6 and 11 above.)

Claim 25 is analyzed and discussed with respect to Claim 8. (See rejection of Claim 8 above.)

Regarding Claims 28-31, Staats teaches in order for the data stream to be transferred to a receiving device, it must comply with the IEEE-1394 Serial Bus Standard such that the receiving device may properly receive the data stream (col. 4, lines 57+). Therefore, a determination is made such that appropriate transmission is performed.

Claim Rejections - 35 U.S.C. § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the

invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 21-22, and 26-27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Staats'821.

Regarding Claim 21, Staats'821 does not specifically disclose the predetermined rate is 29.97 frames per second. However, it is notoriously well known in the art to transmit signal conforming to standard television signals (29.97 frames per second). By performing this method allows for images to be seen on a monitor desirably. Therefore, it would have been obvious to one having ordinary skill in the art to have the predetermined rate to be 29.97 frames per second.

Regarding Claim 22, Staats'821 teaches the x packets represent 267 packets and the y packets represent 266 packets as discussed in Claim 1, but fails to specifically disclose the plurality of second frames are 9336 frames and the plurality of second frames are 664 frames. However, this is an obvious matter of design choice by the manufacturer at the time of production to manufacture such values with respect to the transmission scheme, for it does not change the scope of the invention.

Claims 26 and 27 are analyzed and discussed with respect to Claims 1 and 8. Although Staats'821 teaches 267 packets and 266 packets as discussed in Claim 1, Staats'821 fails to specifically disclose the first frames are 9336 frames and second frames are 664 frames. However, this is an obvious matter of design choice by the manufacturer at the time of production to manufacture such values with respect to the transmission scheme, for it does not change the scope of the invention.

Furthermore, Staats'821 does not specifically disclose the predetermined frame rate is 29.97 frames per second. However, it is notoriously well known in the art to transmit signal conforming to standard television signals (29.97 frames per second). By performing this method allows for images to be seen on a monitor desirably. Therefore, it would have been obvious to one having ordinary skill in the art to have the predetermined rate to be 29.97 frames per second.

Claim 32 is analyzed and discussed with respect to Claim 28. (See rejection of Claim 28 above.)

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jacqueline Wilson whose telephone number is (703) 308-5080. The examiner can normally be reached on 8:30am-5:00pm (alternate Fridays off).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Wendy Garber can be reached on (703) 305-4929. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

JBW

05/28/04

WENDY R. GARBER
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2600

PATENT
Atty. Docket No.: SONY-11300

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of	Group Art Unit: 2612
Quan Vu et al.	Examiner: Wilson, J.
Serial No. 09/249,642	
Filed: February 12, 1999	**PRELIMINARY AMENDMENT**
For: **METHOD OF AND APPARATUS FOR GENERATING A PRECISE FRAME RATE IN DIGITAL VIDEO TRANSMISSION FROM A COMPUTER SYSTEM TO A DIGITAL VIDEO DEVICE**	162 N. Wolfe Rd. Sunnyvale, CA 94086 (408) 530-9700 Customer Number: 28960

Commissioner for Patents
P.O. Box 1450
Alexandria VA 22313-1450

Sir:

Please amend the subject application as follows:

AMENDMENTS

Amendments to the Claims are reflected in the listing of claims which begins on page 2 of this paper.

Remarks/Arguments begin on page 8 of this paper.

CERTIFICATE OF MAILING (37 CFR§ 1.8(a))

I hereby certify that this paper (along with any referred to as being attached or enclosed) is being deposited with the U.S. Postal Service on the date shown below with sufficient postage as first class mail in an envelope addressed to the: Commissioner for Patents, P.O. Box 1450 Alexandria, VA 22313-1450

HAVERSTOCK & OWENS LLP.

Date: 4/30/04 By:

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (currently amended) A method of transmitting information from a source device at a predetermined rate, the method comprising:
   a. calculating a ratio of first data blocks to second data blocks to achieve the predetermined rate;
   b. forming x number of the first data blocks wherein each of the first data blocks contains n units of data;
   ~~b.~~ c. forming y number of the second data blocks wherein each of the second data blocks contains m units of data, and further wherein m is not equal to n; and
   ~~c.~~ d. combining x number of first data blocks and y number of second data blocks into a data stream to achieve the predetermined rate, wherein the first data blocks and the second data blocks are of a same type and have same characteristics and further wherein the x number of first data blocks are evenly distributed among the y number of second data blocks thereby forming a repeating pattern of the first data blocks and the second data blocks within the data stream.

2. (original) The method according to claim 1 further comprising transmitting the data stream from the source device at the predetermined rate.

3. (previously cancelled)

4. (original) The method according to claim 1 wherein the data stream is digital video data.

5. (original) The method according to claim 1 wherein n, m, x, and y are integer values.

6. (currently amended) A method of transmitting information from a source device to a receiving device, the method comprising:
   a. calculating a ratio of first frames to second frames to achieve a predetermined frame rate;

b. forming x number of the first frames wherein each of the first frames contains n units of data;

~~b.~~ c. forming y number of the second frames wherein each of the second frames contains m units of data, and further wherein m is not equal to n;

~~c.~~ d. combining x number of the first frames and y number of the second frames into a stream of frames to achieve ~~a~~ the predetermined frame rate by evenly distributing the x number of the first frames among the y number of the second frames thereby forming a repeating pattern of the first frames and the second frames within the stream of frames; and

~~d.~~ e. transmitting the stream of frames from the source device to the receiving device; wherein the first frames and the second frames are of a same type and have same characteristics.

7. (original) The method according to claim 6 wherein n, m, x, and y are integer values.

8. (original) The method according to claim 6 further comprising receiving the stream of frames from the network by the receiver at a predetermined frame rate and wherein the data stream conforms to standards of an IEEE 1394-1995 network.

9. (previously cancelled)

10. (original) The method according to claim 6 wherein the stream of frames conforms to standards of an IEEE 1394-1995 network.

11. (original) The method according to claim 6 wherein the source device and the receiving device are coupled together within a network.

12. (original) The method according to claim 11 wherein the network is an IEEE 1394-1995 network.

13. (currently amended) A source device for transmitting information at a predetermined frame rate, the source device comprising a controller for calculating a ratio of first frames to second frames to achieve the predetermined frame rate and generating a data stream containing a plurality of the first frames each including x packets of data and a plurality of the second frames

each including y packets of data to achieve the predetermined frame rate, wherein the data stream is transmitted at the predetermined frame rate and y is not equal to x and further wherein the first frames and the second frames are of a same type and have same characteristics and the x number of first data blocks are evenly distributed among the y number of second data blocks thereby forming a repeating pattern of the first frames and the second frames within the data stream.

14. (original) The source device according to claim 13 wherein x and y are integer values.

15. (original) The source device according to claim 13 further comprising an interface coupled to the controller and configured for connecting to a network.

16. (original) The source device according to claim 15 wherein the network is a IEEE 1394-1995 network.

17. (currently amended) A system for transmitting information at a predetermined frame rate, the system comprising:

a. a source device for calculating a ratio of first frames to second frames to achieve the predetermined frame rate and generating a data stream containing a plurality of the first frames each including x packets of data and a plurality of the second frames each including y packets of data to achieve the predetermined frame rate and y is not equal to x, wherein the first frames and the second frames are of a same type and have same characteristics and the x number of first data blocks are evenly distributed among the y number of second data blocks thereby forming a repeating pattern of the first frames and the second frames within the data stream; and

b. a remote receiver coupled to the source device and configured to receive the data stream at the predetermined frame rate.

18. (original) The system according to claim 17 wherein x and y are integer values.

19. (previously amended) The system according to claim 17 wherein the source device is a computer system.

20. (original) The system according to claim 17 wherein the remote receiver is a digital video camera.

21. (original) The system according to claim 17 wherein the predetermined frame rate is 29.97 frames per second.

22. (original) The system according to claim 17 wherein the plurality of first frames are 9336 frames, x packets represent 267 packets, the plurality of second frames are 664 frames, and y packets represent 266 packets.

23. (original) The system according to claim 17 wherein the data stream conforms to standards of an IEEE 1394-1995 network.

24. (original) The system according to claim 17 further comprising a network coupled between the source device and the remote receiver and configured to transmit the data stream.

25. (original) The system according to claim 24 wherein the network is an IEEE 1394-1995 network.

26. (currently amended) A system for transmitting information at a predetermined frame rate equal to 29.97 frames per second within an IEEE 1394 network of devices, the system comprising:

a. a source device for <u>calculating a ratio of first frames to second frames to achieve the predetermined frame rate and</u> generating a data stream containing 9336 first frames, each including 267 packets of data, and 664 second frames, each including 266 packets of data, to achieve the predetermined frame rate of 29.97 frames per second, wherein the first frames and the second frames are of a same type and have same characteristics and the x number of first data blocks are evenly distributed among the y number of second data blocks thereby forming a repeating pattern of first frames and second frames within the data stream; and

b. a remote receiver coupled to the source device by the IEEE 1394 network of devices, wherein the remote receiver receives the data stream from the source device at the predetermined frame rate.

27. (currently amended) A method of transmitting information from a source device to a receiving device over an IEEE 1394 network of devices, the method comprising:

a. calculating a ratio of first frames to second frames;

b. forming 9336 first frames wherein each of the first frames contains 267 packets of data;

~~b.~~ c. forming 664 second frames wherein each of the second frames contains 266 packets of data;

~~c.~~ d. combining the 9336 first frames and the 664 second frames into a stream of frames to achieve a predetermined frame rate of 29.97 frames per second by evenly distributing the second frames among the first frames thereby forming a repeating pattern of the first frames and the second frames within the stream of frames; and

~~d.~~ e. transmitting the stream of frames from the source device to the receiving device over the IEEE 1394 network of devices;

wherein the first frames and the second frames are of a same type and have same characteristics.

Please add the following new claims:

28. (new) The method according to claim 1 wherein the predetermined rate is determined by a receiving device which receives the data stream.

29. (new) The method according to claim 6 wherein the predetermined frame rate is determined by the receiving device.

30. (new) The source device according to claim 13 wherein the predetermined rate is determined by a receiving device which receives the data stream.

31. (new) The system according to claim 17 wherein the predetermined frame rate is determined by the remote receiver.

32. (new) The system according to claim 26 wherein the predetermined frame rate is determined by the remote receiver.

REMARKS

Applicants respectfully request further examination and reconsideration in view of the above amendments and the arguments set forth fully below. Claims 1, 2, 4-8, and 10-27 were pending. Within the Office Action, Claims 1, 2, 4-8, and 10-27 have been rejected. By the above amendment, Claims 1, 6, 13, 17, 26 and 27 have been amended and new Claims 28-32 have been added. Accordingly, Claims 1, 2, 4-8 and 10-32 are currently pending in this application.

Rejections Under 35 U.S.C. § 102

Within the previous Office Action, Claims 1, 2, 4-8, 10-20 and 23-25 were rejected under 35 U.S.C. §102 (e) as being anticipated by U.S. Patent No. 6,373,821 to Staats (hereinafter "Staats"). Staats teaches a method for setting a time stamp in the SYT field of packet headers for IEEE-1394 devices. Staats teaches stamping isochronous data packets with a presentation time stamp value determined according to a computed packet rate for the data. Staats teaches that a computed packet rate for the data can be a non-integer value. To achieve this non-integer value, Staats teaches using a data stream command language. The data stream command language is a set of commands that control data flow into or out of a data stream. Staats teaches that the data stream command language jump commands are used to allow a transmitter to send a frame with a different number of packets. Staats does not teach forming x number of first data blocks each containing n units of data, forming y number of second data blocks each containing m units of data and combining x number of first data blocks and y number of second data blocks into a data stream to achieve the predetermined rate.

Within the previous Office Action, in the response to arguments section, it is stated that Staats specifically teaches that the transmitter needs to send 266 packets and sometimes send 267 packets. It is then stated that this is synonymous to the claimed first and second data blocks with n and m units of data. The applicants respectfully disagree. Staats teaches that sometimes the transmitter will need to send 266 packets/frame and sometimes 267 packets/frame. [Staats, col. 6, lines 7-16] Staats does not teach forming x number of first data blocks each containing n units of data, forming y number of second data blocks each containing m units of data and combining x

number of first data blocks and y number of second data blocks into a data stream to achieve the predetermined rate. Staats also does not teach evenly distributing the x number of first data blocks among the y number of second data blocks. As described above, Staats only teaches that sometimes the transmitter will need to send 266 packets/frame and sometimes 267 packets/frame.

Within the previous Office Action, also in the response to arguments section, it is stated that the term "sometimes" is believed to be used to teach that 267 packets are placed in the stream at certain times in order to produce the NTSC compatible signal but not placed in the stream as constantly as 266 packets, thus, maintaining a proper stream of data. It is further stated within the Office Action that this is still synonymous to the claimed first and second data blocks with n and m units of data and to the claimed "evenly distributed." The applicants respectfully disagree. There is no teaching within Staats regarding evenly distributing the x number of first data blocks among the y number of second data blocks. Further, there is no hint, teaching or suggestion, within Staats to even support an obviousness rejection of evenly distributing the x number of first data blocks among the y number of second data blocks.

Within the previous Office Action, it is stated that uniformity in the data stream is inherent in the system of Staats. The applicants respectfully disagree. Staats teaches that the system determines when the driver should be notified to vary the default number of packets per frame, on a frame by frame basis. [Staats, col. 8, lines 21-67] Specifically, Staats teaches calculating a delta value for each frame, such that if the delta value is equal to or greater than one, a frame with 267 packets is transmitted and if the delta value is less than one, a frame with 266 packets is transmitted. [Staats, col. 8, lines 54-61] In the example, illustrated in Table 1 of Staats, three frames with 266 packets are followed by one frame of 267 packets, then one frame of 266 frames and then one frame of 267 packets. Accordingly, as shown in this example, within the patent itself Staats does not teach that the frames with 267 packets are **evenly distributed** with the frames with 266 packets within the data stream, as claimed in the present claims. Further, Staats teaches calculating an SYT value for a current frame and then calculating the delta value for the current frame, on a frame by frame basis. **Staats does not teach evenly distributing x number of first data blocks among y number of second data blocks thereby forming a repeating pattern of the first data blocks and the second data blocks within the data stream.**

Within the previous Office Action, it is further argued that over the time period of one hour, the sequence of data blocks will eventually repeat itself. The applicants respectfully disagree. *The applicants also question where support within Staats for such a conclusion can be found.* ***No where is this taught, hinted at or suggested in the actual teachings of Staats.*** As discussed above, Staats teaches calculating the delta value for each frame and making a determination based on the delta value as to whether a frame with 267 packets or a frame with 266 packets should be transmitted. Based on this scheme taught by Staats, one cannot **assume** that a pattern will repeat over time, no matter how long the time period. In fact, Staats goes further and even describes what happens when a cycle is missed. The repeating sequence argument, made within the Office Action, does not take such events into account and thus fails when the actual teachings of Staats are analyzed.

Staats teaches determining, on a frame by frame basis, what number of packets will be included within a frame. Staats does not teach **evenly distributing** x number of first data blocks among the y number of second data blocks. Within the previous Office Action, an opinion about what is inherent in the teachings of Staats is all that is used to support a rejection of the claims. However, this opinion is not based on or supported by the actual teachings of Staats, but instead is based on conjecture and examples of how a system of Staats is **assumed** to operate. This can not form a proper basis of a rejection of the claims of the present invention.

There is nothing in the teachings of Staats that supports an anticipation rejection under 35 U.S.C. § 102 of claims with such limitations. **Staats simply does not teach evenly distributing the x number of first data blocks among the y number of second data blocks.** Staats also does not teach that this even distribution forms a repeating pattern of the first data blocks and the second data blocks within the data stream. **As described above, Staats teaches determining the number of packets per frame on a frame by frame basis using a calculated delta value.** Also, as described above, the example shown in Table 1 of Staats does not show an even distribution or a repeating pattern. Further, there is no hint, teaching or suggestion to even warrant an obviousness determination. To do so would be to impermissibly use hindsight to make a rejection based on obviousness. The Court of Appeals for the Federal Circuit has stated that "it is impermissible to use the claimed invention as an instruction manual or 'template' to piece together the teachings of the prior art so that the claimed invention is rendered obvious." In Re Fritch, 972 F.2d, 1260, 1266, 23 USPQ2d 1780, 1784 (Fed. Cir. 1992). Based on the teachings of Staats, it would not have been obvious to evenly distribute the x number of first data blocks among the y number of second data blocks thereby forming a repeating pattern of the first

data blocks and the second data blocks within the data stream. To conclude that this is obvious based on the teachings of Staats, is to use hindsight based on the teachings of the present invention and to read much more into the teachings of this cited reference than its actual teachings. This is simply not permissible based on the directive from the Court of Appeals for the Federal Circuit. All that Staats teaches is that "[t]o achieve an overall $M_{av} = 266.5$, sometimes the transmitter will need to send 266 packets/frame and sometimes 267 packets/ frame." [Staats, col. 6, lines 12-14] There is no hint, teaching or suggestion within Staats to justify a conclusion that it is obvious to **evenly distribute** the 266 packets/frame among the 267 packets/frame.

In contrast to the teachings of Staats, the present invention is directed to a method of and apparatus for transmitting an isochronous video stream of data at a particular frame rate from a source device to a receiving device. The source device preferably determines a proper ratio of data packets versus video frames in response to the particular frame rate required and a cycle time for isochronous data. This proper ratio of data packets versus video frames rarely computes to an integer result. Accordingly, once the proper ratio of data packets versus video frames is determined, the source device preferably generates two groups of frames. A first group contains an integer value of packets nearest to and above the desired overall average ratio of data packets versus video frames. The source device also generates a second group of frames where each frame from this second group contains an integer value of packets nearest to and below the ratio of packets versus video frames. In order to achieve the desired frame rate, the source device generates a frame ratio containing a specific number of frames from the first group and the second group and forms the isochronous stream of video data. Accordingly, the frames from the first group and the frames from the second group are of a same type and have the same characteristics. The source device serially generates each of the frames in an order including a combination of the first group of frames and the second group of frames to achieve the overall desired average frame ratio. The source device then transmits the resulting isochronous video stream of data to the receiving device at the desired frame rate. As described above, Staats does not teach forming x number of first data blocks each containing n units of data, forming y number of second data blocks each containing m units of data and combining x number of first data blocks and y number of second data blocks into a data stream to achieve the predetermined rate. **Staats also does not teach evenly distributing the x number of first data blocks among the y number of second data blocks thereby forming a repeating pattern of the first data blocks and the second data blocks within the data stream.**

As described above, Staats teaches determining, on a frame by frame basis, what number of packets will be included within a frame. The teachings of Staats require this determination to be made for every frame. In contrast to the teachings of Staats, the present invention calculates a ratio of first frames and second frames in response to the particular frame rate. Staats does not teach calculating a ratio of first frames and second frames. As discussed above, Staats teaches determining a number of packets for each frame, on a frame by frame basis. In order to further the prosecution of the present application, this limitation has been added to each of the independent claims.

The independent Claim 1 is directed to a method of transmitting information from a source device at a predetermined rate. The method of Claim 1 comprises calculating a ratio of first data blocks to second data blocks to achieve the predetermined rate, forming x number of the first data blocks wherein each of the first data blocks contains n units of data, forming y number of the second data blocks wherein each of the second data blocks contains m units of data, and further wherein m is not equal to n and combining x number of first data blocks and y number of second data blocks into a data stream to achieve the predetermined rate. Claim 1 includes the further limitation that the first data blocks and the second data blocks are of a same type and have same characteristics. Claim 1 also includes the limitation that the x number of first data blocks are **evenly distributed** among the y number of second data blocks thereby forming a repeating pattern of the first data blocks and the second data blocks within the data stream. As described above, Staats does not teach forming x number of first data blocks each containing n units of data, forming y number of second data blocks each containing m units of data and combining x number of first data blocks and y number of second data blocks into a data stream to achieve the predetermined rate. As also described above, Staats does not teach that the x number of first data blocks are **evenly distributed** among the y number of second data blocks thereby forming a repeating pattern of the first data blocks and the second data blocks within the data stream. Further, Staats does not teach calculating a ratio of first data blocks to second data blocks to achieve the predetermined rate. As discussed above, Staats teaches determining a number of packets for each frame, on a frame by frame basis. For at least these reasons, the independent Claim 1 is allowable over the teachings of Staats.

Claims 2, 4 and 5 are all dependent upon the independent Claim 1. As discussed above, the independent Claim 1 is allowable over the teachings of Staats. Accordingly, Claims 2, 4 and 5 are all also allowable as being dependent upon an allowable base claim.

The independent Claim 6 is directed to a method of transmitting information from a source device to a receiving device. The method of Claim 6 comprises calculating a ratio of first frames to second frames to achieve a predetermined frame rate, forming x number of the first frames wherein each of the first frames contains n units of data, forming y number of the second frames wherein each of the second frames contains m units of data and further wherein m is not equal to n, combining x number of the first frames and y number of the second frames into a stream of frames to achieve the predetermined frame rate by **evenly distributing** the x number of the first frames among the y number of the second frames thereby forming a repeating pattern of the first frames and the second frames within the stream of frames and transmitting the stream of frames from the source device to the receiving device. Claim 6 includes the further limitation that the first frames and the second frames are of a same type and have same characteristics. As described above, Staats does not teach forming x number of first frames wherein each of the first frames contains n units of data, forming y number of second frames wherein each of the second frames contains m units of data and combining x number of the first frames and y number of the second frames into a stream of frames to achieve a predetermined rate. As discussed above, Staats also does not teach **evenly distributing** the x number of first frames among the y number of second frames thereby forming a repeating pattern of the first frames and the second frames within the stream of frames. Further, Staats does not teach calculating a ratio of first frames to second frames to achieve a predetermined frame rate. As discussed above, Staats teaches determining a number of packets for each frame, on a frame by frame basis. For at least these reasons, the independent Claim 6 is allowable over the teachings of Staats.

Claims 7, 8 and 10-12 are all dependent upon the independent Claim 6. As discussed above, the independent Claim 6 is allowable over the teachings of Staats. Accordingly, Claims 7, 8 and 10-12 are each also allowable as being dependent upon an allowable base claim.

The independent Claim 13 is directed to a source device for transmitting information at a predetermined frame rate. The source device of Claim 13 comprises a controller for calculating a ratio of first frames to second frames to achieve the predetermined frame rate and generating a data stream containing a plurality of the first frames each including x packets of data and a plurality of the second frames each including y packets of data to achieve the predetermined frame rate, wherein the data stream is transmitted at the predetermined frame rate and y is not equal to x. Claim 13 includes the further limitation that the first frames and the second frames are of a same type and have same characteristics. It is also specified in Claim 13 that the x number of first data blocks are **evenly distributed** among the y number of second data blocks

thereby forming a repeating pattern of the first frames and the second frames within the data stream. As described above, Staats does not teach generating a data stream including a plurality of first frames each including x packets of data and a plurality of second frames each including y packets of data to achieve the predetermined frame rate. As also described above, Staats does not teach that the x number of first data blocks are **evenly distributed** among the y number of second data blocks thereby forming a repeating pattern of the first frames and the second frames within the data stream. Further, Staats does not teach calculating a ratio of first frames to second frames to achieve a predetermined frame rate. As discussed above, Staats teaches determining a number of packets for each frame, on a frame by frame basis. For at least these reasons, the independent Claim 13 is allowable over the teachings of Staats.

Claims 14-16 are all dependent upon the independent Claim 13. As discussed above, the independent Claim 13 is allowable over the teachings of Staats. Accordingly, Claims 14-16 are each also allowable as being dependent upon an allowable base claim.

The independent Claim 17 is directed to a system for transmitting information at a predetermined frame rate. The system of Claim 17 comprises a source device for calculating a ratio of first frames to second frames to achieve the predetermined frame rate and generating a data stream containing a plurality of first frames each including x packets of data and a plurality of second frames each including y packets of data to achieve the predetermined frame rate and y is not equal to x, wherein the first frames and the second frames are of a same type and have same characteristics and the x number of first data blocks are **evenly distributed** among the y number of second data blocks thereby forming a repeating pattern of the first frames and the second frames within the data stream, and a remote receiver coupled to the source device and configured to receive the data stream at the predetermined frame rate. As described above, Staats does not teach generating a data stream containing a plurality of first frames each including x packets of data and a plurality of second frames each including y packets of data to achieve the predetermined frame rate and y is not equal to x. As discussed above, Staats also does not teach that the x number of first data blocks are **evenly distributed** among the y number of second data blocks thereby forming a repeating pattern of the first frames and the second frames within the data stream. Further, Staats does not teach calculating a ratio of first frames to second frames to achieve a predetermined frame rate. As discussed above, Staats teaches determining a number of packets for each frame, on a frame by frame basis. For at least these reasons, the independent Claim 17 is allowable over the teachings of Staats.

Claims 18-20 and 23-25 are all dependent on the independent Claim 17. As discussed above, the independent Claim 17 is allowable over the teachings of Staats. Accordingly, Claims 18-20 and 23-25 are each also allowable as being dependent upon an allowable base claim.

Rejections Under 35 U.S.C. § 103

Within the previous Office Action, Claims 21, 22, 26 and 27 were rejected under 35 U.S.C. §103 (a) as being unpatentable over Staats. Claims 21 and 22 are both dependent on the independent Claim 17. As discussed above, the independent Claim 17 is allowable over the teachings of Staats. Accordingly, Claims 21 and 22 are both also allowable as being dependent upon an allowable base claim.

The independent Claim 26 is directed to a system for transmitting information at a predetermined frame rate equal to 29.97 frames per second within an IEEE 1394 network of devices. The system of Claim 26 comprises a source device for calculating a ratio of first frames to second frames to achieve the predetermined frame rate and generating a data stream containing 9336 first frames, each including 267 packets of data, and 664 second frames, each including 266 packets of data, to achieve the predetermined frame rate of 29.97 frames per second, wherein the first frames and the second frames are of a same type and have same characteristics and the x number of first data blocks are **evenly distributed** among the y number of second data blocks thereby forming a repeating pattern of first frames and second frames within the data stream, and a remote receiver coupled to the source device by the IEEE 1394 network of devices, wherein the remote receiver receives the data stream from the source device at the predetermined frame rate. As recognized with the Office Action, Staats fails to disclose a data stream containing 9336 first frames and 664 second frames. It is stated in the Office Action that this is an obvious matter of design choice. The applicants respectfully disagree. Staats cites an NTSC compatible device with 266.973 data packets per frame, as an example. [Staats, col. 5, line 64 - col. 6, line 12] However, as discussed above, Staats only teaches that sometimes the transmitter will need to send 266 packets/frame and sometimes 267 packets/frame. As evidence that the limitation of a data stream containing 9336 first frames, each including 267 packets of data, and 664 second frames, each including 266 packets of data, is not an obvious design choice, even though Staats cites as an example an NTSC compatible device with 266.973 data packets per frame, Staats does not describe such a data stream with 9336 first frames and 664 second frames. As discussed

above, Staats only teaches that sometimes the transmitter will need to send 266 packets/frame and sometimes 267 packets/frame. Accordingly, Staats does not teach or make obvious a source device for generating a data stream containing 9336 first frames, each including 267 packets of data, and 664 second frames, each including 266 packets of data. As discussed above, Staats also does not teach or make obvious **evenly distributing** the x number of first frames among the y number of second frames thereby forming a repeating pattern of first frames and second frames within the data stream. Further, Staats does not teach calculating a ratio of first frames to second frames to achieve a predetermined frame rate. As discussed above, Staats teaches determining a number of packets for each frame, on a frame by frame basis. For at least these reasons, the independent Claim 26 is allowable over the teachings of Staats.

The independent Claim 27 is directed to a method of transmitting information from a source device to a receiving device over an IEEE 1394 network of devices. The method of Claim 27 comprises calculating a ratio of first frames to second frames, forming 9336 first frames wherein each of the first frames contains 267 packets of data, forming 664 second frames wherein each of the second frames contains 266 packets of data, combining the 9336 first frames and the 664 second frames into a stream of frames to achieve a predetermined frame rate of 29.97 frames per second by **evenly distributing** the second frames among the first frames thereby forming a repeating pattern of the first frames and the second frames within the stream of frames and transmitting the stream of frames from the source device to the receiving device over the IEEE 1394 network of devices, wherein the first frames and the second frames are of a same type and have same characteristics. As described above, Staats does not teach or make obvious forming 9336 first frames wherein each of the first frames contains 267 packets of data, forming 664 second frames wherein each of the second frames contains 266 packets of data and combining the 9336 first frames and the 664 second frames into a stream of frames to achieve a predetermined frame rate of 29.97 frames per second. As also described above, Staats does not teach **evenly distributing** the second frames among the first frames thereby forming a repeating pattern of the first frames and the second frames within the stream of frames. Further, Staats does not teach calculating a ratio of first frames to second frames to achieve a predetermined frame rate. As discussed above, Staats teaches determining a number of packets for each frame, on a frame by frame basis. For at least these reasons, the independent Claim 27 is allowable over the teachings of Staats.

For the reasons given above, Applicants respectfully submit that all of the claims are in a condition for allowance, and allowance at an early date would be appreciated. Should the Examiner have any questions or comments, they are encouraged to call the undersigned at (408) 530-9700 to discuss the same so that any outstanding issues can be expeditiously resolved.

Respectfully submitted,
HAVERSTOCK & OWENS LLP

Dated: April 30, 2004

By: Jonathan O. Owens
Jonathan O Owens
Reg. No. 37,902
Attorneys for Applicant(s)